

**Manuel d'Utilisation**  
**Fascicule U4.5- : Méthode de résolution**  
**Document : U4.51.03**

# Opérateur *STAT\_NON\_LINE*

---

## 1 But

---

Calculer l'évolution mécanique ou thermo-hydro-mécanique couplée, en quasi-statique, d'une structure en non linéaire.

La non linéarité est liée soit au comportement du matériau (par exemple plastique), soit à la géométrie (par exemple en grands déplacements). Pour avoir des détails sur la méthode de résolution employée, on se reportera à la documentation de référence [R5.03.01].

L'évolution peut être étudiée en plusieurs travaux successifs (concept réentrant), soit en poursuite (le dernier instant calculé est l'instant initial du calcul suivant), soit en reprise en partant d'un instant antérieur.

Si le temps nécessaire pour effectuer le calcul n'est pas suffisant, le programme s'interrompt, mais les résultats déjà calculés sont sauvegardés si une base de données a été définie dans le profil d'étude de l'utilisateur. Produit une structure de données de type `evol_noli`.

## Table des matières

1 But .....	1
2 Syntaxe .....	4
3 Opérandes .....	9
3.1 Opérandes MODELE / CHAM_MATER / CARA_ELEM .....	9
3.2 Mot clé EXCIT .....	9
3.2.1 Opérandes CHARGE .....	9
3.2.2 Opérande FONC_MULT .....	10
3.2.3 Opérande TYPE_CHARGE .....	10
3.3 Mot clé VARI_COMM .....	11
3.3.1 Opérande IRR .....	11
3.4 Mot clé ETAT_INIT .....	11
3.4.1 Opérandes SIGM / VARI / DEPL / VARI_NON_LOCAL .....	11
3.4.2 Opérandes EVOL_NOLI .....	11
3.4.3 Opérande NUME_ORDRE / INST / NUME_DIDI .....	12
3.4.4 Opérande INST_ETAT_INIT .....	12
3.4.5 Opérande PRECISION / CRITERE .....	13
3.5 Mot clé INCREMENT .....	13
3.5.1 Opérandes LIST_INST / EVOLUTION .....	13
3.5.2 Opérandes NUME_INST_INIT / INST_INIT / NUME_INST_FIN / INST_FIN .....	14
3.5.3 Opérande PRECISION .....	15
3.5.4 Opérande SUBD_PAS / SUBD_PAS_MINI / COEF_SUBD_PAS_1 .....	15
3.5.5 Opérande OPTI_LIST_INST / NOM_CHAM / NOM_CMP / VALEUR .....	15
3.6 Mot clé NEWTON .....	16
3.6.1 Opérande PREDICTION .....	16
3.6.2 Opérande MATRICE .....	17
3.6.3 Opérande EVOL_NOLI .....	18
3.7 Mot clé RECH_LINEAIRE .....	18
3.7.1 Opérande RESI_LINE_RELA / ITER_LINE_MAXI .....	18
3.7.2 Opérande PAS_MINI_CRIT / ITER_LINE_CRIT .....	18
3.7.3 Opérandes RHO_MIN / RHO_MAX / RHO_EXCL .....	18
3.8 Opérande PARM_THETA .....	19
3.9 Mot clé PILOTAGE .....	19
3.9.1 Opérande TYPE .....	19
3.9.2 Opérandes NOEUD / GROUP_NO .....	21
3.9.3 Opérandes TOUT / MAILLE / GROUP_MA .....	21
3.9.4 Opérande NOM_CMP .....	21
3.9.5 Opérande COEF_MULT .....	21

Titre :           Opérateur *STAT\_NON\_LINE*  
Auteur(s) :    **P. BADEL**

Date :           14/09/04  
Clé :    U4.51.03-G2 Page :   3/30

3.9.6	Opérande ETA_PILO_R_MAX / ETA_PILO_R_MIN.....	22
3.9.7	Opérande ETA_PILO_MAX / ETA_PILO_MIN.....	22
3.9.8	Opérande PROJ_BORNES.....	22
3.9.9	Opérande SELECTION.....	23
3.10	Mot clé SOLVEUR .....	23
3.11	Mot clé CONVERGENCE .....	23
3.11.1	Opérande RESI_GLOB_RELA / RESI_GLOB_MAXI.....	23
3.11.2	Opérande RESI_REFE_RELA.....	24
3.11.3	Opérande ITER_GLOB_MAXI .....	24
3.11.4	Opérande ITER_GLOB_ELAS.....	24
3.11.5	Opérande ARRET.....	24
3.11.6	Opérandes RESI_INTE_RELA / ITER_INTE_MAXI .....	24
3.11.7	Opérande ITER_INTE_PAS .....	25
3.11.8	Opérande RESO_INTE .....	25
3.12	Mot-clé CRIT_FLAMB.....	25
3.13	Mot-clé SENSIBILITE.....	26
3.14	Mot clé ARCHIVAGE .....	26
3.14.1	Opérande LIST_INST / INST / PAS_ARCH .....	26
3.14.2	Opérande PRECISION .....	26
3.14.3	Opérande ARCH_ETAT_INIT / NUME_INIT / DETR_NUME_SUIV.....	26
3.14.4	Opérande CHAM_EXCLU .....	28
3.15	Opérande OBSERVATION.....	28
3.16	Opérande SOLV_NON_LOCAL .....	28
3.17	Opérande LAGR_NON_LOCAL .....	28
3.18	Opérande INFO .....	29
3.19	Opérande TITRE.....	29

## 2 Syntaxe

```

statnl [evol_noli] = STAT_NON_LINE
(
  ◇ reuse =      statnl,                                [evol_noli]
  ◆ MODELE =    mo,                                    [modele]
  ◆ CHAM_MATER = chmat,                                [cham_mater]
  ◇ CARA_ELEM =  carac,                                [cara_elem]
  ◆ EXCIT =      _F (
    ◆ CHARGE =   chi,                                  [char_meca]
    ◇ FONC_MULT = fi,                                  [fonction/formule]
    ◇ TYPE_CHARGE = / 'FIXE_CSTE'                      [DEFAULT]
                      / 'FIXE_PILO'
                      / 'SUIV'
                      / 'DIDI'
    ),
  ◆ | COMP_INCR = _F (voir le document [U4.51.11] ),
    | COMP_ELAS = _F (voir le document [U4.51.11] ),
  ◇ VARI_COMM = _F (
    ◆ / IRRA = irra                                    [evol_varc]
    ),
  ◇ ETAT_INIT = _F (
    ◆ / | SIGM =          sig,                        [cham_elem_SIEF_R]
        | VARI =          vain,                      [carte_SIEF_R]
        | DEPL =          depl,                      [cham_elem_VARI_R]
        | VARI_NON_LOCAL = vanolo,                  [cham_no_DEPL_R]
        | VARI_NON_LOCAL = vanolo,                  [cham_no_VANL_R]
    / EVOL_NOLI =          evol,                      [evol_noli]
    ◇ / NUME_ORDRE=      nuini,                      [I]
      / INST =          instini,                    [R]
    ◇ PRECISION =        / 1.0E-3,                  [DEFAULT]
      / prec,                                                  [R]
    ◇ CRITERE   =        / 'RELATIF',                [DEFAULT]
      / 'ABSOLU',
    ◇ NUME_DIDI =        nudidi,                      [I]
    ◇ INST_ETAT_INIT =    istetaini                  [R]
    ),

```

Titre :           Opérateur *STAT\_NON\_LINE*  
Auteur(s) :     **P. BADEL**

Date :           14/09/04  
Clé :    U4.51.03-G2 Page :   5/30

```
♦ INCREMENT =_F (
    ♦ LIST_INST =                   litps,                   [listr8]

    ◇ EVOLUTION =                   / 'CHRONOLOGIQUE',     [DEFAULT]
                                    / 'RETROGRADE',
                                    / 'SANS',

    ◇ / NUME_INST_INIT =           nuini,                   [I]
      / INST_INIT =                instini,                [R]

    ◇ / NUME_INST_FIN =           nufin,                   [I]
      / INST_FIN =                 instfin,                [R]

    ◇ PRECISION =                  / 1.0E-3,               [DEFAULT]
                                    / prec,                 [R]

    ◇ SUBD_PAS =                   / 1,                     [DEFAULT]
                                    / subpas,               [I]

    ◇ SUBD_PAS_MINI =              submini,                [R]

    ◇ COEF_SUBD_PAS_1 =           / 1.,                    [DEFAULT]
                                    / coefsub,             [R]

    ◇ OPTI_LIST_INST =             / 'INCR_MAXI',           [DEFAULT]

    ◇ NOM_CHAM =                   nomch,                  [Kn]

    ◇ NOM_CMP =                    nomcmp,                 [Kn]

    ◇ VALEUR =                     val                     [R]
    ),

◇ NEWTON =_F (
    ◇ PREDICTION =                 / 'TANGENTE',           [DEFAULT]
                                    / 'ELASTIQUE',
                                    / 'EXTRAPOL',
                                    / 'DEPL_CALCULE',

    ◇ EVOL_NOLI =                   evol_noli,              [evol_noli]

    ◇ MATRICE =                    / 'TANGENTE',           [DEFAULT]
                                    / 'ELASTIQUE'

    ◇ REAC_INCR =                  / 1,                     [DEFAULT]
                                    / mf,                   [I]

    ◇ REAC_ITER =                  / 0,                     [DEFAULT]
                                    / it,                   [I]

    ◇ PAS_MINI_ELAS =              / 0,                     [DEFAULT]
                                    / pasmini,             [I]
    ),
```

Titre :        *Opérateur STAT\_NON\_LINE*  
Auteur(s) :   **P. BADEL**

Date :        14/09/04  
Clé :    *U4.51.03-G2* Page :    6/30

```
◇ RECH_LINEAIRE =_F (
    ◇ RESI_LINE_RELA =           / 1.E-1,           [DEFAULT]
                                   / reslin,           [R]
    ◇ ITER_LINE_MAXI =           / 3                 [DEFAULT]
                                   / itelin            [I]
    ◇ PAS_MINI_CRIT =            / 0.                [DEFAULT]
                                   / pmicri            [R]
    ◇ ITER_LINE_CRIT =           / 20                [DEFAULT]
                                   / itelic            [I]
    ◇ RHO_MIN =                  / 1.E-2             [DEFAULT]
                                   / rmin              [R]
    ◇ RHO_MAX =                  / 1.E+1             [DEFAULT]
                                   / rmax              [R]
    ◇ RHO_EXCL =                 / 9.E-3            [DEFAULT]
                                   / rexc              [R]
    ),
◇ PARM_THETA =                  / 1.,             [DEFAULT]
                                   / theta,            [R]
◇ PILOTAGE =_F (
    ◆ TYPE =                     / 'DDL_IMPO',
                                   / 'LONG_ARC',
                                   / 'ANA_LIM',
                                   / 'DEFORMATION',
                                   / 'PRED_ELAS',

    ◇ / TOUT =                    'OUI',               [DEFAULT]
      / GROUP_MA =                lgrma,               [l_gr_maille]
      / MAILLE =                  lma,                 [l_maille]
    ◇ / NOEUD =                   no,                  [noeud]
      / GROUP_NO =                grno,                [gr_noeud]

    ◇ NOM_CMP =                   nomcmp,               [Kn]

    ◇ COEF_MULT =                 / 1.,                [DEFAULT]
                                   / cmult,              [R]
    ◇ ETA_PILO_R_MAX =            etarmax,              [R]
    ◇ ETA_PILO_R_MIN =            etarmin,              [R]
    ◇ ETA_PILO_MAX =              etamax,               [R]
    ◇ ETA_PILO_MIN =              etamin,               [R]

    ◇ PROJ_BORNES =               / 'OUI'              [DEFAULT]
                                   / 'NON'

    ◇ SELECTION =                 / 'NORM_INCR_DEPL',   [DEFAULT]
                                   / 'ANGL_INCR_DEPL',
                                   / 'RESIDU',

    ),
```

Titre :           Opérateur *STAT\_NON\_LINE*  
Auteur(s) :     **P. BADEL**

Date :           14/09/04  
Clé :    U4.51.03-G2 Page :   7/30

```
◇ 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_GLOB_REFE = resref, [R]  
    ◇ SIGM_REFE = sigref, [R]  
  
    ◇ ITER_GLOB_ELAS = / 25, [DEFAULT]  
                      / maxelas, [I]  
  
    ◇ ITER_GLOB_MAXI = / 10, [DEFAULT]  
                      / maglob, [I]  
  
    ◇ ARRET = / 'OUI', [DEFAULT]  
             / 'NON',  
  
    ◇ RESI_INTE_RELA = / 1.E-6, [DEFAULT]  
                      / resint, [R]  
  
    ◇ ITER_INTE_MAXI = / 10, [DEFAULT]  
                      / iteint, [I]  
  
    ◇ ITER_INTE_PAS = / 0, [DEFAULT]  
                     / itepas, [I]  
  
    ◇ RESO_INTE = / 'IMPLICITE', [DEFAULT]  
                 / 'RUNGE_KUTTA_2',  
                 / 'RUNGE_KUTTA_4',  
    ),  
◇ CRIT_FLAMB =_F (   
    ◇ NB_FREQ = / 3, [DEFAULT]  
               / nbfreq, [I]  
    ◇ CHAR_CRIT = / (-10,10), [DEFAULT]  
                 / intcc,  
    ),  
◇ 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-3, [DEFAULT]  
                  / prec, [R]  
  
    ◇ / ARCH_ETAT_INIT = 'OUI',  
      / NUME_INIT = nuinit, [I]  
  
    ◇ DETR_NUME_SUIV = 'OUI',  
  
    ◇ CHAM_EXCLU = | 'DEPL',  
                   | 'SIEF_ELGA',  
                   | 'VARI_ELGA',  
                   | 'VARI_NON_LOCAL',  
                   | 'LANL_ELGA',  
    ),
```

Titre :        Opérateur *STAT\_NON\_LINE*  
Auteur(s) :   **P. BADEL**

Date :        14/09/04  
Clé :    *U4.51.03-G2* Page :    8/30

```
◇  OBSERVATION =_F ( voir le document [U4.53.01] ),  
  
◇  LAGR_NON_LOCAL =_F (  
  
    ◇  ITER_PRIM_MAXI =      / 10,                [DEFAULT]  
                           / iterprimmax,        [I]  
  
    ◆  RESI_PRIM_ABSO =      resiprimab,          [R]  
  
    ◇  ITER_DUAL_MAXI =      / 50,                [DEFAULT]  
                           / iterdmax,           [I]  
  
    ◆  RESI_DUAL_ABSO =      residabso,           [R]  
  
    ◇  R =                  / 1000.              [DEFAULT]  
                           / rho                  [R]  
    ),  
  
◇  SOLV_NON_LOCAL =_F ( voir le document [U4.50.01] ),  
  
◇  INFO =                  / 1,                  [DEFAULT]  
                           / 2,  
  
◇  TITRE = tx              [Kn]  
  
);
```



## 3 Opérandes

### 3.1 Opérandes MODELE / CHAM\_MATER / CARA\_ELEM

- ◆ **MODELE** = mo
- ◆ **CHAM\_MATER** = chmat
- ◇ **CARA\_ELEM** = carac

Ces mots-clés permettent de renseigner :

- le nom du modèle (mo) dont les éléments font l'objet du calcul mécanique,
- le nom du champ de matériau (chmat) affecté sur le maillage. Attention, toutes les mailles du modèle doivent être associées à un matériau (sinon erreur fatale avec message peu explicite),
- le nom des caractéristiques (carac) des éléments de coque, poutre, tuyau, barre, câble, et éléments discrets affectés sur le modèle mo. Evidemment, ce mot-clé est optionnel : si le modèle ne contient pas de tels éléments, il n'est pas utile ; en revanche, si le modèle contient de tels éléments, il est obligatoire.

### 3.2 Mot clé EXCIT

- ◆ **EXCIT** :

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.

#### 3.2.1 Opérandes CHARGE

- ◆ **CHARGE** : ch<sub>i</sub>

ch<sub>i</sub> est le chargement mécanique (comportant éventuellement l'évolution d'un champ de température) précisé à la i<sup>ème</sup> occurrence de EXCIT.

*Une et une seule charge peut comporter l'évolution d'un champ de température, qui aura précédemment été défini grâce au mot-clé **TEMP\_CALCULEE** de la commande **AFFE\_CHAR\_MECA**.*

**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.*

**Attention :**

*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 ».*

**Attention :**

| Les chargements issus de *AFFE\_CHAR\_CINE* ne sont pas utilisables avec *STAT\_NON\_LINE*.

**3.2.2 Opérande FONC\_MULT**

◇ **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 :

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

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

Par défaut :  $f_i = 1$ .

**Remarque :**

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

**3.2.3 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 'FIXE\_PILLO', le chargement est toujours fixe (indépendant de la géométrie) mais sera piloté grâce au mot clé PILOTAGE [§3.11]. Les charges pilotables doivent être issues d'AFFE\_CHAR\_MECA ou d'AFFE\_CHAR\_MECA\_F et ne pas être affectées du mot clé FONC\_MULT. On ne peut pas piloter les chargements de pesanteur, la force centrifuge, les forces de Laplace, les chargements thermiques ou de déformations initiales ou anélastiques, et les conditions de liaison.

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 :  $u = d$ .

## 3.3 Mots-clés COMP\_INCR et COMP\_ELAS

La syntaxe de ces mots-clés commun à plusieurs commandes est décrite dans le document [U4.51.11].

## 3.4 Mot clé VARI\_COMM

◇ **VARI\_COMM :**

Variables de commandes qui pilotent les lois de comportement (au même titre que la température).

### 3.4.1 Opérande IRRA

◆ **/ IRRA : irr**

Champs d'irradiation.

## 3.5 Mot clé ETAT\_INIT

◇ **ETAT\_INIT :**

Etat initial de référence choisi. Par défaut, tous les champs sont identiquement nuls. Cet état initial peut être défini soit en précisant chaque champ de l'état initial, soit en extraction depuis un concept de type *evol\_noli* préexistant.

La donnée d'un état initial n'a de sens (et n'est donc prise en compte) que pour la partie du domaine traitée en comportement incrémental (**COMP\_INCR**) ; si le comportement est élastique (**COMP\_ELAS**) cela n'a aucune incidence.

**Si l'on veut prendre en compte un état initial en élasticité, c'est le mot clé ELAS situé sous COMP\_INCR qu'il faut utiliser.**

**Remarque :**

*Dans le cas où l'utilisateur a spécifié que le concept résultat est réentrant (par le mot réservé *reuse*), le mot-clé **ETAT\_INIT** est obligatoire.*

### 3.5.1 Opérandes SIGM / VARI / DEPL / VARI\_NON\_LOCAL

◆ /	<b>SIGM</b>	=	<b>sig</b>
	<b>VARI</b>	=	<b>vain</b>
	<b>DEPL</b>	=	<b>depl</b>
	<b>VARI_NON_LOCAL</b>	=	<b>vanolo</b>

Respectivement, champs de contraintes aux points de Gauss, de variables internes aux points de Gauss, de déplacements aux nœuds et de variables non locales aux nœuds (pour des modèles non locaux) pris à l'état initial. Si l'un de ces champs n'est pas précisé, il est pris nul par défaut. Ils peuvent par exemple être issus de la commande **RECU\_CHAMP**, ou bien avoir été lus dans un fichier au format I-DEAS par la commande **LIRE\_RESU** (attention le format MED ne lit que des champs aux nœuds).

### 3.5.2 Opérandes EVOL\_NOLI

**/ EVOL\_NOLI : evol**

Nom du concept de type *evol\_noli* d'où sera extrait l'état initial.

### 3.5.3 Opérande NUME\_ORDRE / INST / NUME\_DIDI

◇        /    NUME\_ORDRE    =    nuini  
          /    INST            =    instini

Extraction de l'état mécanique initial dans *evol* à partir du numéro d'archivage NUME\_ORDRE ou de l'instant d'archivage INST pour effectuer la poursuite du calcul.  
Si NUME\_ORDRE ou INST ne sont pas remplis, on prend le dernier numéro archivé existant dans *evol*.

◇    NUME\_DIDI :    nudidi

Dans le cas de chargements de type DIRICHLET différentiel ('DIDI'), on donne sous NUME\_DIDI le numéro d'archivage de l'état mécanique (déplacement) qui sert de référence pour l'application de ces conditions aux limites (Cf. [§3.2.2]). Par défaut on prend l'état mécanique défini sous NUME\_ORDRE ou INST.

### 3.5.4 Opérande INST\_ETAT\_INIT

◇    INST\_ETAT\_INIT :    istetaini

On peut associer une valeur d'instant *istetaini* à cet état initial.  
Par défaut :

- lorsque l'état initial est défini par la donnée des champs, il n'y a pas d'instant associé.
- lorsque l'état est donné par un concept *evol\_noli*, il s'agit de l'instant dans le précédent calcul (*istetaini* = *instini*).

#### Attention danger :

Dans le cas d'un concept réentrant, *INST\_ETAT\_INIT* permet d'avoir des instants identiques pour des incréments différents (exemple C avec *U2=&U1*). Ce cas de figure pose problème, par exemple, dans *IMPR\_COURBEE* lorsque qu'on veut tracer deux fonctions.

#### A - Exemple simple (par défaut)

```
LIST1 = DEFI_LIST_REEL( DEBUT =0.,  
                        INTERVALLE =_F(JUSQU'A = 4., NOMBRE =4)),  
  
U = STAT_NON_LINE (INCREMENT =_F(LIST_INST =LIST1)) ,  
  
LIST2 = DEFI_LIST_REEL( DEBUT =4.,  
                        INTERVALLE =_F(JUSQU'A = 10., NOMBRE =6)),  
  
U = STAT_NON_LINE ( reuse=U,  
                  INCREMENT =_F(LIST_INST =LIST2),  
                  ETAT_INIT =_F(EVOL_NOLI =U)) ,
```

1<sup>er</sup> STAT\_NON\_LINE : effectue le calcul pour les instants 1, 2, 3 et 4s.

2<sup>nd</sup> STAT\_NON\_LINE : effectue le calcul pour les instants 5, 6, 7, 8, 9 et 10s, l'état initial correspondant au temps 4s.

#### B - Exemple pour montrer l'intérêt de INST\_ETAT\_INIT (deux listes d'instant différentes)

```
LIST1 = DEFI_LIST_REEL( DEBUT =0.,  
                        INTERVALLE =_F(JUSQU'A = 10., NOMBRE =10)),  
  
U = STAT_NON_LINE (INCREMENT =_F(LIST_INST =LIST1)) ,  
  
LIST2 = DEFI_LIST_REEL( DEBUT =20.,  
                        INTERVALLE =_F(JUSQU'A = 30., NOMBRE =10)),
```

Titre :           Opérateur *STAT\_NON\_LINE*  
 Auteur(s) :    **P. BADEL**

Date :           14/09/04  
 Clé :    U4.51.03-G2 Page :   13/30

```
U = STAT_NON_LINE ( reuse=U
                    INCREMENT =_F(LIST_INST =LIST2),
                    ETAT_INIT =_F(EVOL_NOLI =U,
                    INST_ETAT_INIT = 20.)) ,
```

1<sup>er</sup> STAT\_NON\_LINE : effectue le calcul des instants 1 à 10s.

2<sup>nd</sup> STAT\_NON\_LINE : effectue le calcul des instants 21 à 30s, l'état initial correspondant à l'instant t=10s du 1<sup>er</sup> STAT\_NON\_LINE (par défaut INST=10.). Cet état initial correspond pour ce 2<sup>nd</sup> STAT\_NON\_LINE à l'instant t=20s. (INST\_ETAT\_INIT=20.).

### C - Exemple pour montrer l'intérêt de INST\_ETAT\_INIT (pratique quand on fait du cyclique)

```
LIST1 = DEFI_LIST_REEL( DEBUT =0.,
                       INTERVALLE =_F(JUSQU'A = 10., NOMBRE =10)),

U1 = STAT_NON_LINE( INCREMENT =_F( LIST_INST =LIST1)) ,

U2 = STAT_NON_LINE( INCREMENT =_F( LIST_INST =LIST1),
                    ETAT_INIT =_F( EVOL_NOLI =U1,
                    INST_ETAT_INIT = 0.)) ,
```

1<sup>er</sup> STAT\_NON\_LINE : effectue le calcul des instants 1 à 10s.

2<sup>nd</sup> STAT\_NON\_LINE : effectue le calcul des instants 1 à 10s, l'état initial correspondant à l'instant t=10s du 1<sup>er</sup> STAT\_NON\_LINE (par défaut INST=10.). Cet état initial correspond pour ce 2<sup>nd</sup> STAT\_NON\_LINE à l'instant t=0s. (INST\_ETAT\_INIT : 0.).

## 3.5.5 Opérande PRECISION / CRITERE

Cf. [U4.71.00].

## 3.6 Mot clé INCREMENT

### ◆ INCREMENT :

Définit les intervalles de temps pris dans la méthode incrémentale.

Les instants ainsi définis n'ont de sens physique que pour des relations de comportement où le temps intervient explicitement (visco-élastiques ou visco-plastiques par exemple). Dans les autres cas, ils permettent seulement d'indicer les incréments de charge et de paramétrer l'évolution d'un éventuel champ de température.

### 3.6.1 Opérandes LIST\_INST / EVOLUTION

#### ◆ LIST\_INST : litps

Les instants de calcul sont ceux définis dans le concept litps par l'opérateur DEFI\_LIST\_REEL [U4.34.01].

◇ EVOLUTION :    /    'CHRONOLOGIQUE'           [DEFAULT]  
                   /    'RETROGRADE'  
                   /    'SANS'

Le mot clé 'CHRONOLOGIQUE' permet de vérifier si la liste d'instants donnée par l'utilisateur est strictement croissante (si non message d'erreur).

Le mot clé 'RETROGRADE' permet d'inverser la liste d'instants donnée par l'utilisateur et de vérifier qu'après cette opération, elle est bien strictement décroissante.

Il n'y a pas de vérification lorsqu'on précise une évolution 'SANS'.

**3.6.2 Opérandes NUME\_INST\_INIT / INST\_INIT / NUME\_INST\_FIN / INST\_FIN**

```

◇ / NUME_INST_INIT = nuini
  / INST_INIT      = instini

```

L'instant initial du calcul (qui donc n'est pas (re)calculé) est désigné soit par sa valeur (INST\_INIT), soit par son numéro d'ordre dans la liste d'instantanés litps (NUME\_INST\_INIT). Pour pouvoir accéder par valeur, il est nécessaire que la liste soit ordonnée (EVOLUTION : 'CHRONOLOGIQUE' ou 'RETROGRADE').

En l'absence des mots clés INST\_INIT ou NUME\_INST\_INIT, le défaut est calculé de la manière suivante :

- si un état initial est précisé (opérande ETAT\_INIT) et s'il définit un instant correspondant (par EVOL\_NOLI ou INST\_ETAT\_INIT) alors l'instant initial est celui défini par l'état initial,
- s'il n'y a pas d'état initial (opérande ETAT\_INIT) ou qu'il ne définit pas d'instant correspondant (les champs sont donnés dans ETAT\_INIT sans préciser INST\_ETAT\_INIT), alors on prend le premier instant de la liste d'instantanés litps (NUME\_INST\_INIT : 0), ou le dernier lorsque l'évolution est rétrograde.

```

◇ / NUME_INST_FIN = nufin
  / INST_FIN      = instfin

```

L'instant final (dernier pas calculé) est désigné de la même manière que l'instant initial (soit NUME\_INST\_FIN, soit INST\_FIN), sauf qu'il n'est pas possible de faire référence à l'instant de l'état initial.

**Attention :** avec une évolution RETROGRADE, INST\_INIT > INST\_FIN.

**A - Exemple simple (par défaut)**

```

LIST = DEFI_LIST_REEL( DEBUT =0.,
                       INTERVALLE =_F(JUSQU'A= 10., NOMBRE =10)),

U = STAT_NON_LINE ( INCREMENT =_F ( LIST_INST =LIST,
                                     INST_FIN =4.)) ,

U = STAT_NON_LINE( reuse=U,
                  INCREMENT =_F ( LIST_INST =LIST),
                  ETAT_INIT =_F (EVOL_NOLI :U)) ,

```

1<sup>er</sup> STAT\_NON\_LINE : effectue le calcul pour les instants 1, 2, 3 et 4s.

2<sup>nd</sup> STAT\_NON\_LINE : effectue le calcul pour les instants 5, 6, 7, 8, 9 et 10s, l'état initial correspondant au temps 4s. (par défaut INST\_INIT=INST\_ETAT\_INIT=INST=4.).

**B - Exemple pour montrer l'intérêt de INST\_INIT**

```

LIST = DEFI_LIST_REEL( DEBUT =0.,
                       INTERVALLE =_F (JUSQU'A = 10., NOMBRE =10)),

U = STAT_NON_LINE ( INCREMENT =_F( LIST_INST = LIST,
                                     INST_FIN = 4.)) ,

U = STAT_NON_LINE ( reuse = U,
                  INCREMENT =_F ( LIST_INST =LIST,
                                  INST_INIT =8.),
                  ETAT_INIT =_F ( EVOL_NOLI =U)) ,

```

1<sup>er</sup> STAT\_NON\_LINE : effectue le calcul des instants 1 à 4s.

2<sup>nd</sup> STAT\_NON\_LINE : effectue le calcul pour les instants 9 et 10s (ne fait rien pour t=5, 6, 7 et 8s), l'état initial correspondant au temps t=4s (par défaut INST=4.).

### 3.6.3 Opérande PRECISION

◇ PRECISION : prec Cf. [U4.71.00]

### 3.6.4 Opérande SUBD\_PAS / SUBD\_PAS\_MINI / COEF\_SUBD\_PAS\_1

◇ SUBD\_PAS = subpas  
◇ SUBD\_PAS\_MINI = submini  
◇ COEF\_SUBD\_PAS\_1 = coefsub

Permet de réaliser un redécoupage automatique du pas de temps lorsque l'algorithme de Newton ne converge pas.

Le pas de temps est redécoupé en subpas sous pas. Par défaut il n'y a pas de redécoupage (subd\_pas : 1). La subdivision automatique s'arrête lorsque les nouveaux pas créés sont plus petits que SUBD\_PAS\_MINI. Les nouveaux pas créés sont de taille identique, excepté le premier qui est égal à cette taille multipliée par COEF\_SUBD\_PAS\_1 (par défaut 1). Ceci permet de mieux prendre en compte les problèmes de décharge de la structure (changement de matrice tangente) sans utiliser la matrice élastique (PREDICTION : 'ELASTIQUE' ou MATRICE : 'ELASTIQUE' sous l'opérande NEWTON).

Lorsqu'un pas de temps a été redécoupé plusieurs fois (appelons n le nombre de fois où l'on a procédé à une subdivision du même pas), le pas suivant est automatiquement subdivisé (n-1) fois, ceci pour éviter, en cas de convergence difficile de tenter un pas de temps trop important.

**Remarque concernant le mot clé DECOUPE sous SOLVEUR :**

*Lors de calcul de flambage élastoplastique, il peut arriver que la matrice tangente du système soit singulière au cours des itérations de Newton. En redécoupant le pas de temps, on peut passer ces points durs. Sous l'opérande SOLVEUR, le mot clé DECOUPE sous STOP\_SINGULIER sert à gérer ces points durs. Il est alors nécessaire de renseigner les mots clés relatifs au redécoupage pour que la méthode DECOUPE soit activée.*

### 3.6.5 Opérande OPTI\_LIST\_INST / NOM\_CHAM / NOM\_CMP / VALEUR

◇ OPTI\_LIST\_INST = 'INCR\_MAXI' [DEFAULT]  
◇ NOM\_CHAM = 'TEMP' [DEFAULT]  
◇ NOM\_CMP = 'TEMP' [DEFAULT]  
◇ VALE = vale

Ces opérandes n'ont d'intérêt que lorsqu'on réalise un calcul thermomécanique. Permet de créer si besoin une nouvelle liste de pas de temps mécanique de sorte que entre chaque incrément de temps, l'incrément de température soit inférieur à une valeur donnée par l'utilisateur et renseignée par le mot clé VALE.

La création de cette nouvelle liste se fait de la façon suivante :

- Liste d'instants initial en mécanique :  $T_i$
- Liste d'instants thermique :  $\theta$
- Nouvelle liste d'instants mécanique final à créer si besoin :  $T_f$
- On insère entre chaque intervalle de la liste initiale mécanique  $T_i$ , les instants thermiques inclus dans cet intervalle. On récupère alors pour chaque intervalle une liste d'instants  $\tau = [\tau_0, \tau_1, \tau_2, \dots, \tau_N]$
- Construction de la liste final  $T_f$ 
  - Initialisation :  $\tau_f = \tau_0$
  - 1<sup>er</sup> Test :  
Si  $T(\tau_j) - T(\tau_f) > \text{valeur}$  avec  $T(t)$  la température au temps  $t$  et  $\tau_f$  le dernier instant inséré dans la nouvelle liste  $T_f$ , alors on garde dans la nouvelle liste  $T_f$ , l'instant  $\tau_{j-1}$
  - 2<sup>ème</sup> Test :  
Si  $T(\tau_j) - T(\tau_{j-1}) > \text{valeur}$  alors on redécoupe uniformément cet intervalle de façon à satisfaire la condition sur l'incrément de température.

**Exemple :** SI  $T(\tau) = [T(\tau_1) = 20^\circ\text{C}, T(\tau_2) = 30^\circ\text{C}, T(\tau_3) = 55^\circ\text{C}, T(\tau_4) = 65^\circ\text{C}]$  avec  $VALE = 15^\circ\text{C}$

Initialisation :  $\tau_f = \tau_1$

Intervalle 1 :

1<sup>er</sup> Test = 2<sup>ème</sup> Test :  $T(\tau_2) - T(\tau_1) = 10^\circ\text{C} < 15$  donc on  $T_f = [\tau_1]$

Intervalle 2 :

1<sup>er</sup> Test :  $T(\tau_3) - T(\tau_f) = 35^\circ\text{C} > 15$  donc on a  $T_f = [\tau_1, \tau_2]$  et  $\tau_f = \tau_2$

2<sup>ème</sup> Test :  $T(\tau_3) - T(\tau_2) = 25^\circ\text{C} > 15$  donc on  $T_f = [\tau_1, \tau_2, T_3 \text{ tel que } T(T_3) = 42.5^\circ\text{C}, \tau_3]$  et  $\tau_f = \tau_3$

Intervalle 3 :

1<sup>er</sup> Test = 2<sup>ème</sup> Test :  $T(\tau_4) - T(\tau_3) = 10^\circ\text{C} < 15^\circ\text{C}$

d'où la liste finale suivante :

$T_f = [\tau_1, \tau_2, T_3 \text{ tel que } T(T_3) = 42.5^\circ\text{C}, \tau_3, \tau_4]$

## 3.7 Mot clé NEWTON

### ◇ NEWTON :

Précise les caractéristiques de la méthode de résolution du problème incrémental non linéaire (méthode de NEWTON-RAPHSON).

### 3.7.1 Opérande PREDICTION

◇ PREDICTION = / 'TANGENTE'  
/ 'ELASTIQUE'  
/ 'EXTRAPOL'  
/ 'DEPL\_CALCULE'

La phase de prédiction (Cf. [R5.03.01]) a pour but de calculer une estimation du champ de déplacements afin de permettre à la méthode de NEWTON de converger plus rapidement.

Lorsque le mot clé est absent, c'est la matrice tangente en vitesse (option RIGI\_MECA\_TANG dans le fichier .mess) qui est utilisée si l'on a choisi pour la méthode de NEWTON une MATRICE: 'TANGENTE', et c'est la matrice élastique (option RIGI\_MECA dans le fichier .mess) qui est utilisée si on a choisi MATRICE: 'ELASTIQUE'.

/ 'TANGENTE'

On utilise la matrice tangente du problème en vitesse (option RIGI\_MECA\_TANG dans le fichier .mess).

/ 'ELASTIQUE'

On utilise la matrice élastique (option RIGI\_MECA dans le fichier .mess).

/ 'EXTRAPOL'

On calcule l'estimation de l'incrément de déplacement à partir de l'incrément total obtenu comme solution au pas de temps précédent (pondéré par le rapport des pas de temps). On projette cette estimation sur l'ensemble des champs cinématiquement admissibles (i.e. satisfaisant les conditions aux limites de DIRICHLET) selon la norme donnée par la matrice élastique, qui doit donc être calculée. Cette fonctionnalité est intéressante dans le cas de l'utilisation de schémas d'intégration locale explicite de type RUNGE-KUTTA qui ne fournissent pas de matrice tangente : dans ce cas la méthode de NEWTON utilise une matrice élastique, mais le nombre d'itérations nécessaires peut être élevé. L'utilisation de l'extrapolation peut améliorer les performances.

/ 'DEPL\_CALCULE'

Permet de proposer comme déplacement pour la prédiction à chaque pas de temps, le déplacement donné par une histoire mécanique précisée sous le mot clé EVOL\_NOLI ([§3.8.3]).



**Utilité :**

- supposons qu'on réalise un premier calcul avec un maillage grossier. On souhaite réaliser le même calcul mais sur un maillage plus fin. On peut supposer que la solution en déplacement pour ce second calcul n'est pas éloignée de celle du premier calcul et donc qu'une bonne prédiction du déplacement pour ce second calcul est la projection des déplacements du calcul 1 sur les nœuds du nouveau maillage (la projection des déplacements sur le nouveau maillage doit être réalisée préalablement avec l'opérateur PROJ\_CHAMP [U4.72.05]). Ce mot clé permet de réaliser ce mode de prédiction.
- cela permet de réduire la place mémoire et de conserver ces résultats en vue d'une poursuite ultérieure. Pour un gros calcul, on peut stocker uniquement les déplacements à tous les instants aux formats IDEAS ou MED dans IMPR\_RESU. Si on veut recalculer les contraintes et variables internes, on fait un LIRE\_RESU au format adéquate puis on utilise DEPL\_CALCULE avec ITER\_GLOB\_MAXI : 0 (on effectue une seule itération) et ARRET :NON (il n'y a pas convergence, on ne vérifie pas l'équilibre). Il est toutefois nécessaire pour des raisons de syntaxe de donner un chargement (éviter les chargements dirichlet qui imposent une résolution linéaire) ainsi qu'un critère de convergence, même si ces informations ne sont pas prises en compte.

### 3.7.2 Opérande MATRICE

```
◇ MATRICE =  
/ 'TANGENTE'  
◇ REAC_INCR = / 1 [DEFAULT]  
/ mf  
◇ REAC_ITER = / 0 [DEFAULT]  
/ it
```

La matrice utilisée pour les itérations globales de la méthode est la matrice tangente [R5.03.01] qui est réévaluée tous les *mf* incréments de temps (*mf* positif ou nul) et toutes les *it* itérations de NEWTON pour un incrément de temps donné (précisément aux itérations de numéro *it*, *2it*, *3it*...). Donc à la première itération de NEWTON, on ne réassemble la matrice tangente que si *it* vaut 1 : sinon on garde la matrice utilisée dans la phase de prédiction. Par convention si *it* vaut 0 la matrice n'est pas réévaluée durant tout le pas de temps.

```
◇ PAS_MINI_ELAS = 0. [DEFAULT]  
pasmini [I]
```

Permet de passer de la matrice tangente à la matrice de décharge (i.e en considérant que les non linéarités n'évoluent pas) lorsque le pas de temps est ou devient (par le redécoupage) inférieur à *pasmini*. Cette matrice de décharge est la matrice élastique pour les modèles de comportement de type plastique ; pour les modèles d'endommagement elle s'identifie à la matrice sécante.

Comme la convergence avec la matrice élastique est plus lente que celle avec la matrice tangente, le mot clé ITER\_GLOB\_ELAS sous le mot clé facteur CONVERGENCE permet de définir un nombre d'itérations maximal spécifique à l'utilisation de la matrice élastique et différent de celui associé à l'utilisation de la matrice tangente.

**Utilité :**

Cette option peut être utile lorsque le redécoupage automatique du pas de temps (cf. [§ 3.7.4]) ne suffit pas à faire converger un calcul. Par exemple, dans le cas de lois adoucissantes, la matrice tangente peut devenir singulière et il vaut donc mieux utiliser la matrice élastique pour converger.

```
/ 'ELASTIQUE'
```

La matrice utilisée correspond au calcul élastique : elle n'est évaluée qu'une fois à l'instant initial, en début d'algorithme.

Cette matrice "élastique" est calculée en utilisant le module d'YOUNG donné sous le mot clé ELAS de l'opérateur DEFI\_MATERIAU, et non pas la pente à l'origine de la courbe de traction donnée sous le mot clé TRACTION (et qui sert, elle, dans l'expression de la relation de comportement).

### 3.7.3 Opérande EVOL\_NOLI

◇ EVOL\_NOLI : evol\_noli

Nom du concept de type evol\_noli qui servira dans la prédiction par DEPL\_CALCULE.

## 3.8 Mot clé RECH\_LINEAIRE

RECH\_LINEAIRE :

La recherche linéaire peut permettre d'améliorer la convergence de la méthode de Newton (Cf. [R5.03.01] pour plus de détails).

**Attention :**

*Il est déconseillé d'utiliser la recherche linéaire avec les déformations GREEN\_GR pour les modélisations COQUE\_3D.*

### 3.8.1 Opérande RESI\_LINE\_RELA / ITER\_LINE\_MAXI

◇ RESI\_LINE\_RELA =    / 1.E-1            [DEFAULT]  
                              / reslin  
◇ ITER\_LINE\_MAXI =    / 3                [DEFAULT]  
                              / itelin

Ce sont les paramètres de la recherche linéaire. On donne le nombre d'itérations maximum itelin à effectuer et la précision reslin à atteindre pour réaliser la convergence de la recherche linéaire. Il est conseillé de ne pas utiliser la recherche linéaire avec du contact.

Il n'est pas nécessaire de spécifier une précision ni un nombre d'itérations très élevés, la pratique montrant que 2 ou 3 itérations de recherche linéaire sont suffisantes. On peut donc se contenter de demander 3 itérations avec la précision par défaut.

### 3.8.2 Opérande PAS\_MINI\_CRIT / ITER\_LINE\_CRIT

◇ PAS\_MINI\_CRIT    =    / 0.                [DEFAULT]  
                              / pmicri            [R]  
◇ ITER\_LINE\_CRIT    =    / 20               [DEFAULT]  
                              / itelic            [I]

Lors de pas de temps où la convergence est délicate, on peut vouloir augmenter le nombre maximum d'itérations de recherche linéaire. C'est ce que permettent les mots-clés PAS\_MINI\_CRIT et ITER\_LINE\_CRIT. Quand le pas de temps (directement fixé par l'utilisateur ou conséquence de découpages de pas de temps) devient inférieur à la valeur pmicri, le nombre d'itérations de recherche de recherche linéaire passe de itelin (renseigné par ITER\_LINE\_MAXI) à itelic (renseigné par ITER\_LINE\_MAXI)

### 3.8.3 Opérandes RHO\_MIN / RHO\_MAX / RHO\_EXCL

◇ RHO\_MIN            =    / 1.E-2            [DEFAULT]  
                              / rmin             [R]  
◇ RHO\_MAX            =    / 1.E+1            [DEFAULT]  
                              / rmax             [R]  
◇ RHO\_EXCL           =    / 9.E-3            [DEFAULT]  
                              / rexc             [R]

Ces mots-clés fixent l'intervalle  $I$  de la recherche linéaire, sous la forme :  
 $I = [r_{\min}, r_{\max}] - [-rexc, rexc]$ .

## 3.9 Opérande PARM\_THETA

◇ PARM\_THETA = / 1. [DEFAULT]  
/ theta

Pour les modélisations THM, l'argument *theta* est le paramètre de la thêta-méthode utilisée pour résoudre les équations évolutives de thermique et d'hydraulique (Cf. [R5.03.60] pour plus de détails). Sa valeur doit être comprise entre 0 (méthode explicite) et 1 (méthode totalement implicite).

Pour les lois de comportements ROUSS\_VISC, ASSE\_COMBU, ZIRC\_CYRA2 et ZIRC\_EPRI, l'argument *theta* sert à l'intégration de la loi de comportement (pour le modèle ASSE\_COMBU, il sert à intégrer la loi de Lemaitre en 1D). Il peut prendre les valeurs 0.5 (semi-implicite) ou 1 (implicite).

## 3.10 Mot clé PILOTAGE

◇ PILOTAGE :

Lorsque l'intensité  $\eta$  d'une partie du chargement n'est pas connue a priori (chargement dit de référence défini dans AFPE\_CHAR\_MECA ou AFPE\_CHAR\_MECA\_F avec charge de type *FIXE\_PILLO*), le mot clé *PILOTAGE* permet de piloter ce chargement par l'intermédiaire d'un nœud (ou groupe de nœud) sur lequel on peut imposer différents modes de pilotage (mot clé *TYPE*).

**Attention :**

Avec *FIXE\_PILLO*, on ne peut pas utiliser pour le chargement de référence le mot clé *FONCT\_MULT*.

**Attention :**

Lorsque le chargement de référence est défini par *AFPE\_CHAR\_MECA\_F*, ce chargement peut être fonction des variables d'espace mais pas du temps.

**Attention :**

Le mot clé *PILOTAGE* est interdit avec le contact.

### 3.10.1 Opérande TYPE

◇ TYPE : / 'DDL\_IMPO'  
/ 'LONG\_ARC'  
/ 'ANA\_LIM'  
/ 'DEFORMATION'  
/ 'PRED\_ELAS'

C'est le type de pilotage effectué. Cinq modes de pilotage sont disponibles (Cf. [R5.03.80] pour plus de détails) :

/ 'DDL\_IMPO'

Permet d'imposer une valeur donnée d'incrément de déplacement (une seule composante *i* possible) en un unique nœud *no* (ou d'un groupe de nœuds ne comportant qu'un seul nœud). À chaque incrément de temps, on cherche l'amplitude  $\eta$  du chargement de référence qui permettra de satisfaire la relation incrémentale suivante :

$$c_{mult} \Delta u_i(no) = \Delta t$$

## / 'LONG\_ARC'

Permet de piloter l'intensité  $\eta$  du chargement de référence par la longueur (abscisse curviligne) de la réponse en déplacement d'un groupe de nœuds (à utiliser par exemple lorsqu'on veut contrôler le flambement d'une éprouvette). On vérifie la relation suivante :

$$C_{mult} \|\Delta u\| = \Delta t \text{ avec } \|\Delta u\| = \sqrt{\sum_n \sum_c \Delta U_{n,c}^2}$$

où  $n$  sont les nœuds du pilotage et  $c$  les composantes du déplacement des nœuds considérés. Même si le groupe de nœud du pilotage est réduit à un seul nœud, il faut quand même utiliser GROUP\_NO.

## / 'ANA\_LIM'

Ce mode de pilotage est spécifique au calcul de charge limite (loi NORTON\_HOFF) par approche cinématique ( cf. [R7.07.01] pour plus de détail). Si  $F$  désigne le chargement assemblé piloté, TYPE\_CHARGE = 'FIXE\_PILLO', alors la fonction de pilotage s'écrit simplement :

$$P(U) = F.U = 1$$

Excepté pour le calcul de charge limite, cette fonctionnalité ne présente pas d'intérêt *a priori*.  
Pour ce mode de pilotage, aucun autres mots clés n'est à préciser.

### Remarque :

*L'utilisation de lois de comportement adoucissantes peut conduire à des snap backs brutaux qui rendent délicat le déroulement du calcul. Les deux modes de pilotage suivants y remédient (Cf. [R5.03.80] pour plus de détail).*

## / 'DEFORMATION'

DEFORMATION garantit qu'au moins un point de Gauss de la structure voit sa déformation évoluer de façon monotone. On vérifie la relation :

$$C_{mult} \max_{\text{point de Gauss}} \left( \frac{\varepsilon^-}{\|\varepsilon^-\|} \Delta \varepsilon \right) = \Delta t$$

Ce mode de pilotage est valable pour toutes les lois de comportement y compris en grandes déformations SIMO\_MIEHE.

## / 'PRED\_ELAS'

PRED\_ELAS assure qu'au moins un point de Gauss de la structure sorte du seuil d'élasticité linéarisé  $f_{\text{préd-élas}}$  d'une quantité  $\Delta T/C_{mult}$ . On vérifie la relation :

$$C_{mult} \max_{\text{point de Gauss}} (f_{\text{préd-élas}}) = \Delta t$$

Ce mode de pilotage est valable uniquement pour les lois ENDO\_FRAGILE (avec la version locale et les deux versions non locales), ENDO\_ISOT\_BETON (avec la version locale et la version non locale), BARENBLATT et BETON\_DOUBLE\_DP.

### Utilisation – Attention :

Lorsqu'on veut utiliser ces deux derniers modes de pilotage, il est indispensable de faire un premier STAT\_NON\_LINE sans le mot clé PILOTAGE pour amorcer le problème et obtenir un état initial  $\varepsilon^-$  différent de zéro (sinon division par zéro pour le pilotage par incrément de déformation). On effectue après une reprise à partir de cet état initial non nul et on utilise le pilotage.  
De plus, la résolution des deux équations précédentes permet d'obtenir l'intensité du chargement inconnue. Dans certains cas, la résolution de ces équations peut conduire à plusieurs solutions pour l'intensité. On choisit alors toujours la solution qui est la plus proche de  $\varepsilon^-$ . C'est pourquoi, lorsqu'on veut imposer un chargement alterné, on est obligé à chaque changement de signe du chargement de réaliser un premier STAT\_NON\_LINE sans le mot clé PILOTAGE afin d'obtenir un état initial  $\varepsilon^-$  de

traction ou de compression. On effectue ensuite un second *STAT\_NON\_LINE* en poursuite à partir de l'état initial précédent avec le mot clé *PILOTAGE*.

**Remarque :**

| *DEFORMATION* et *PRED\_ELAS* ne sont pas disponibles pour les éléments de structures.

### 3.10.2 Opérandes *NOEUD* / *GROUP\_NO*

◇ / *NOEUD* = *no*  
/ *GROUP\_NO* = *grno*

On donne le nom du nœud ou le nom de groupe de nœuds sur lequel on va imposer le pilotage. A n'utiliser qu'avec '*DDL\_IMPO*' ou '*LONG\_ARC*'.

Pour '*DDL\_IMPO*', si on utilise l'opérande *GROUP\_NO*, le groupe de nœuds en question ne doit contenir qu'un seul nœud. Pour '*LONG\_ARC*', on utilise uniquement *GROUP\_NO* (qui peut éventuellement ne contenir qu'un seul nœud).

### 3.10.3 Opérandes *TOUT* / *MAILLE* / *GROUP\_MA*

◇ / *TOUT* = '*OUI*' [DEFAULT]  
/ *GROUP\_MA* = *lgrma*  
/ *MAILLE* = *lma*

On donne les mailles ou groupes de mailles servant à piloter le calcul. A n'utiliser qu'avec *DEFORMATION* ou *PRED\_ELAS*. Intéressant pour alléger la résolution des équations de ces trois modes de pilotages.

### 3.10.4 Opérande *NOM\_CMP*

◇ *NOM\_CMP* : *nomcmp*

C'est le nom de la composante (correspondant au degré de liberté *i*) utilisée pour le pilotage ('*DX*' par exemple). A n'utiliser qu'avec '*DDL\_IMPO*' ou '*LONG\_ARC*'.

### 3.10.5 Opérande *COEF\_MULT*

◇ *COEF\_MULT* : *cmult*

C'est la valeur (notée  $c_{mult}$  dans la formule de définition) par laquelle on multiplie le degré de liberté utilisé pour le pilotage. Par défaut, cette valeur vaut 1. A ne pas utiliser avec *ANA\_LIM*.

**Exemple avec *DDL\_IMPO* :**

Supposons que l'on veut connaître la charge limite d'une structure.

Le chargement imposé sur la structure est la pression d'intensité inconnue ( $P=\eta \cdot$  valeur de référence  $P_x$ ) sur le groupe de maille A. Pour trouver la charge limite  $P_{limite}$ , on va piloter le déplacement du nœud NO1. On veut que le déplacement final suivant *x* de ce nœud soit égal à 2. (soit d'après la liste d'instantanés des pas de 0.2, soit un coefficient  $cmult=1/0.2=5$ .)

```
PRESSION = AFFE_CHAR_MECA ( PRES =( GROUP_MA =A, PX = 1.0) ) ,

LIST = DEFI_LIST_REEL ( DEBUT =0.,
                        INTERVALLE =_F(JUSQU'A = 10, NOMBRE =10),

RESU = STAT_NON_LINE( EXCIT = _F( CHARGE = PRESSION,
                                TYPE_CHARGE = 'FIXE_PILO' ) ,
                        PILOTAGE =_F( TYPE = 'DDL_IMPO' ,
                                NOEUD = NO1 ,
                                NOM_CMP = 'DX' ,
                                COEF_MULT = 5. ) ) ,
```

Dans le fichier.resu, la valeur de  $\eta$  sera affichée à chaque instant du calcul. Pour connaître la charge limite, il suffit de faire  $P_{\text{limite}} = \eta * P_x$ . (Ici  $P_x$  vaut 1 donc on a directement la charge limite). Si on impose sur la structure une pression  $P$  proche de la charge limite sans utiliser le pilotage, le calcul ne convergera pas si on est proche de la charge limite.

### 3.10.6 Opérande ETA\_PILO\_R\_MAX / ETA\_PILO\_R\_MIN

- ◇ **ETA\_PILO\_R\_MAX** = **etarmax**, [R]
- ◇ **ETA\_PILO\_R\_MIN** = **etarmin**, [R]

Ces deux mots-clés permettent de préciser l'intervalle de valeurs de pilotage attendues. Le principe de fonctionnement est le suivant : à chaque itération de Newton, si l'on trouve des valeurs de pilotage dans l'intervalle  $[\text{etar min}, \text{etar max}]$ , toutes les valeurs de pilotage en dehors de cet intervalle ne sont pas considérées. En revanche, si aucune valeur de pilotage n'est trouvée dans cette intervalle, toutes les valeurs de pilotage sont conservées.

Si on ne précise pas de valeurs, c'est  $-\infty$  pour **etarmin** et  $+\infty$  pour **etarmax**.

Une utilisation possible de cet intervalle est le suivant. on désire par exemple, piloter une pression à quelque part sur la structure et on s'attend à garder cette pression positive. En fixant **etarmin** à 0, cela permet de ne conserver que les valeurs de pilotage positives, si on trouve au moins une valeur de pilotage positive lors de la résolution du pilotage.

### 3.10.7 Opérande ETA\_PILO\_MAX / ETA\_PILO\_MIN

- ◇ **ETA\_PILO\_MAX** : **etamax**

Arrêt du calcul lorsque le paramètre de pilotage atteint la valeur donnée **etamax**.

- ◇ **ETA\_PILO\_MIN**: **etamin**

Permet d'interrompre le calcul lorsque le paramètre **ETA\_PILOTAGE** atteint cette valeur minimale **etamin** (pour des modèles adoucissants, permet de stopper le calcul lorsque la structure est suffisamment adoucie).

#### Attention :

| Avec la loi *ENDO\_ISOT\_BETON*, ces deux mots clés sont obligatoires.

### 3.10.8 Opérande PROJ\_BORNES

- ◇ **PROJ\_BORNES** = / 'OUI' [DEFAULT]  
/ 'NON'

En cas de dépassement de l'intervalle (**etamin**, **etamax**), l'utilisateur peut indiquer s'il veut projeter la valeur de pilotage sur (**etamin**, **etamax**).

Avec **PROJ\_BORNE**='OUI', la projection sera effectuée (si  $\text{eta} > \text{etamax} \rightarrow \text{eta} = \text{etamax}$ ; si  $\text{eta} < \text{etamin} \rightarrow \text{eta} = \text{etamin}$ ), ce qui permet, en cas de convergence d'arrêter le calcul précisément sur **etamin** ou **etamax**.

Avec **PROJ\_BORNE**='NON', on ne fait rien, donc le calcul s'arrêtera, en cas de convergence, avec une valeur supérieure à **etamax** ou inférieure à **etamin**.

### 3.10.9 Opérande SELECTION

```

◇ / SELECTION = / 'NORM_INCR_DEPL', [DEFAULT]
                  / 'ANGL_INCR_DEPL',
                  / 'RESIDU',

```

Cet opérande permet de sélectionner la méthode permettant de choisir la valeur de pilotage dans le cas où plusieurs solutions sont fournies par la résolution de pilotage.

'NORM\_INCR\_DEPL' permet de sélectionner la valeur de pilotage par la plus petite norme de l'incrément de déplacement sur le pas de temps considéré.

'ANGL\_INCR\_DEPL' permet de sélectionner la valeur de pilotage par le plus petit angle entre le déplacement obtenu pour le pas de temps courant et le déplacement obtenu pour le pas de temps précédent.

'RESIDU' permet de sélectionner la valeur de pilotage conduisant au plus petit résidu.

### 3.11 Mot clé SOLVEUR

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

### 3.12 Mot clé CONVERGENCE

◇ CONVERGENCE :

Si aucun des deux opérandes suivants n'est présent, alors tout se passe comme si :  
RESI\_GLOB\_RELA = 1.E-6.

#### 3.12.1 Opérande RESI\_GLOB\_RELA / RESI\_GLOB\_MAXI

◇ | RESI\_GLOB\_RELA = resrel

L'algorithme continue les itérations globales tant que :

$$\max_{i=1,\dots,nb\_ddl} |F_i^n| > \text{resrel} \max |\mathbf{L}|$$

où  $F^n$  est le résidu de l'itération  $n$  et  $\mathbf{L}$  le vecteur du chargement imposé et des réactions d'appuis (Cf. [R5.03.01] pour plus de détails).

Lorsque le chargement et les réactions d'appui deviennent nuls, c'est-à-dire lorsque  $\mathbf{L}$  est nul (par exemple dans le cas d'une décharge totale), on passe du critère de convergence relatif au critère de convergence absolu RESI\_GLOB\_MAXI. Cette opération est transparente pour l'utilisateur (message d'alarme émis dans le fichier .mess). Lorsque le vecteur  $\mathbf{L}$  redevient différent de zéro, on repasse automatiquement au critère de convergence relatif RESI\_GLOB\_RELA.

Si cet opérande est absent, le test est effectué avec la valeur par défaut, sauf si RESI\_GLOB\_MAXI est présent.

◇ | RESI\_GLOB\_MAXI = resmax

L'algorithme continue les itérations globales tant que :

$$\max_{i=1,\dots,nb\_ddl} |F_i^n| > \text{resmax}$$

où  $F^n$  est le résidu de l'itération  $n$  (Cf. [R5.03.01] pour plus de détails).

Si cet opérande est absent, le test n'est pas effectué.

Si RESI\_GLOB\_RELA et RESI\_GLOB\_MAXI sont présents tous les deux, les deux tests sont effectués.

## 3.12.2 Opérande RESI\_REFE\_REL

```
| RESI_REFE_REL = resref, [R]
◇ SIGM_REFE = sigref, [R]
```

Cet opérande conduit à estimer la convergence de l'algorithme de Newton de la manière suivante (Cf. RX.XXX pour plus de détails). A partir de la contrainte de référence *sigref*, on calcule une référence de résidu *Fref* (un vecteur de même longueur que le vecteur résidu). La convergence sera réalisée si et seulement si :

$$\forall i \in [1, \dots, nb\_ddl] \quad |F_i^n| < resref \ F_i^{ref}$$

## 3.12.3 Opérande ITER\_GLOB\_MAXI

```
◇ ITER_GLOB_MAXI = / 10 [DEFAULT]
                  / maglob
```

Nombre d'itérations maximum effectué pour résoudre le problème global à chaque instant (10 par défaut). Ce test est toujours effectué.

## 3.12.4 Opérande ITER\_GLOB\_ELAS

```
◇ ITER_GLOB_ELAS = / 25 [DEFAULT]
                  / maxelas
```

Nombre d'itérations maximum effectué avec la matrice élastique lorsqu'on utilise le mot clé PAS\_MINI\_ELAS du mot clé facteur NEWTON (voir [§3.8.2]). pour résoudre le problème global à chaque instant (25 par défaut).

On rappelle que PAS\_MINI\_ELAS permet de passer de la matrice tangente à la matrice élastique lorsque le pas de temps est ou devient (par le redécoupage) inférieur à une certaine valeur précisée sous PAS\_MINI\_ELAS.

## 3.12.5 Opérande ARRET

```
◇ ARRET =
/ 'OUI' [DEFAULT]
```

Si un des critères de convergence globale choisis n'est pas vérifié après *maglob* itérations, alors le programme s'arrête (les résultats précédents sont sauvegardés).

```
/ 'NON'
```

Si *maglob* est insuffisant pour vérifier les critères de convergence donnés par l'utilisateur, on passe quand même à l'instant suivant. Utilisation à éviter.

## 3.12.6 Opérandes RESI\_INTE\_REL / ITER\_INTE\_MAXI

```
◇ RESI_INTE_REL = / 1.E-6 [DEFAULT]
                  / resint
◇ ITER_INTE_MAXI = / 10 [DEFAULT]
                  / iteint
```

Dans la plupart des relations de comportement, une équation non linéaire ou un système non linéaire doivent être résolus localement (en chaque point de GAUSS). Ces opérandes (résidu et nombre maximum d'itérations dites internes) sont utilisés pour tester la convergence de cet algorithme itératif de résolution. Pour plus de détails, se reporter à la documentation de référence, par exemple au document [R5.03.02]. Ces opérandes sont **inutiles** avec les comportements ELAS, VMIS\_CINE\_LINE, VMIS\_ECMI\_LINE, VMIS\_ECMI\_TRAC, VMIS\_ISOT\_LINE, VMIS\_ISOT\_TRAC, BARENBLATT, NORTON\_HOFF, DIS\_CONTACT, DIS\_CHOC, ARME, ASSE\_CORN, DIS\_GOUJ2E\_PLAS, DIS\_GOUJ2E\_ELAS, VMIS\_ASYM\_LINE, GRILLE\_ISOT\_LINE, GRILLE\_CINE\_LINE, GRILLE\_PINTO\_MEN, PINTO\_MENEGOTTO, GRANGER\_FP et GRANGER\_FP\_V (hors contrainte plane), BAZANT\_FD et toutes les relations META\_XXX.



### 3.12.7 Opérande ITER\_INTE\_PAS

```

◇ ITER_INTE_PAS = 0 [DEFAULT]
                  itepas

```

Permet de redécouper localement le pas de temps pour faciliter l'intégration de la relation de comportement aux points de GAUSS (pour les relations de CHABOCHE, VISCTAHERI, ROUSS\_PR, ROUSS\_VISC, CJS et BETON\_DOUBLE\_DP). Si itepas vaut 0, 1 ou -1 il n'y a pas de redécoupage. Si itepas est positif, on redécoupe systématiquement le pas de temps localement en itepas petits pas de temps avant d'effectuer l'intégration de la relation de comportement. Si itepas est négatif, le redécoupage en |itepas| petits pas de temps n'est effectué qu'en cas de non convergence locale.

### 3.12.8 Opérande RESO\_INTE

```

◇ RESO_INTE = / 'IMPLICITE' [DEFAULT]
               / 'RUNGE_KUTTA_2'
               / 'RUNGE_KUTTA_4'

```

Permet de préciser le type de schéma d'intégration pour résoudre le système d'équations non linéaires formé par les équations constitutives des modèles de comportement à variables internes :

- le modèle POLY\_CFC est traité uniquement par le schéma explicite de RUNGE-KUTTA d'ordre 2,
- les deux modèles VMIS\_POU\_LINE et VMIS\_POU\_FLEJOU peuvent être traités par les deux schémas IMPLICITE et RUNGE\_KUTTA\_4,
- les autres modèles utilisent le schéma IMPLICITE.

### 3.13 Mot-clé CRIT\_FLAMB

```

◇ CRIT_FLAMB =_F (
                  ◇ NB_FREQ = / 3, [DEFAULT]
                      / nbfreq, [I]
                  ◇ CHAR_CRIT = / (-10,10), [DEFAULT]
                      / intcc,
                  ),

```

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  $\det(K^T - \lambda K^g) = 0$ .  $K^T$  est la matrice tangente cohérente à cet instant.  $K^g$  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 (cd MODE\_ITER\_SIMULT). 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 déplacement et les grandes déformations GREEN(\_GR) ou SIMO\_MIEHE, on résout  $\det(K^T - \lambda Id) = 0$  car  $K^T$  contient alors  $K^g$  (et éventuellement  $K^p$ ).

Le critère est alors un critère d'instabilité : quand  $\lambda$  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.

## 3.14 Mot-clé SENSIBILITE

La syntaxe de ce mot clé commun à plusieurs commandes est décrite dans le document [U4.50.02].

## 3.15 Mot clé ARCHIVAGE

◇ **ARCHIVAGE =**

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.

Remarque : en présence de contact, on ne peut pas archiver plus de 99 999 instants de calculs

### 3.15.1 Opérande LIST\_INST / INST / PAS\_ARCH

◇ / `'LIST_INST'` = `list_r8`  
/ `'INST'` = `l_r8`  
/ `'PAS_ARCH'` = `npas`

La désignation des instants à stocker est effectuée soit par une liste d'instants (`list_r8` ou `l_r8`) à condition que l'évolution soit ordonnée (`EVOLUTION` : `CHRONOLOGIQUE` ou `RETROGADE`, cf [§3.6.1]) ou alors par une fréquence d'archivage (tous les `npas` de temps).

En l'absence de ces mots clés tous les pas de temps sont archivés.

Deux remarques :

- le dernier pas de calcul est toujours stocké pour pouvoir effectuer une reprise,
- si on emploie un accès par liste d'instants, alors les instants de calculs nouvellement créés par redécoupage automatique du pas de temps ne sont pas archivés

### 3.15.2 Opérande PRECISION

◇ **PRECISION =** `prec`

Cf. [U4.71.00]

### 3.15.3 Opérande ARCH\_ETAT\_INIT / NUME\_INIT / DETR\_NUME\_SUIV

◇ / `'ARCH_ETAT_INIT'` = `'NON'` [DEFAULT]  
`'OUI'`

Uniquement pour un *concept non réentrant* sinon message d'erreur. Permet d'imposer l'archivage de l'état initial dans le numéro d'ordre 0 (intéressant lorsque l'état initial provient d'un autre `STAT_NON_LINE`. Permet d'avoir le 1<sup>er</sup> point sur une courbe).

/ `'NUME_INIT'` = `nuinit`

Uniquement pour un *concept réentrant* sinon message d'erreur. Permet de préciser à partir de quel numéro d'ordre on archive.

Par défaut :

- si l'état initial n'est pas fixé par le concept calculé, il s'agit du dernier numéro d'ordre +1 (exemple A),
- si le concept calculé coïncide avec le concept qui fixe l'état initial, il s'agit du numéro d'ordre +1 sous `ETAT_INIT` (exemple B et C).

◇ DETR\_NUME\_SUIV = 'NON' [DEFAULT]  
'OUI'

Cette opération peut conduire à écraser des numéros d'ordre préexistants : le mot clé DETR\_NUME\_SUIV confirme cette destruction, tandis que son absence met fin au calcul.

## A - Exemple simple

```
LIST = DEFI_LIST_REEL( DEBUT =0.,  
                      INTERVALLE =_F(JUSQU'A =5., NOMBRE =5)),  
  
U1 = STAT_NON_LINE( INCREMENT =_F( LIST_INST =LIST,  
                                   INST_FIN =3.)) ,  
  
U2 = STAT_NON_LINE( INCREMENT =_F(LIST_INST =LIST)) ,  
  
U2 = STAT_NON_LINE( reuse=U2,  
                  ETAT_INIT =_F(EVOL_NOLI =U1),  
                  INCREMENT =_F(LIST_INST =LIST),  
                  ARCHIVAGE =_F(LIST_INST = LIST)) ,
```

Le résultat final pour l'archivage de U2 est le suivant :

numéro d'archivage	: 1	2	3	4	5	6	7
instants correspondants	: 1.	2.	3.	4.	5.	4.	5.

## B - Exemple simple

```
LIST = DEFI_LIST_REEL( DEBUT =0.,  
                      INTERVALLE =_F(JUSQU'A =10., NOMBRE =5)),  
  
U2 = STAT_NON_LINE( INCREMENT =_F(LIST_INST =LIST)) ,  
  
&U2 = STAT_NON_LINE( reuse=U2,  
                  ETAT_INIT =_F( EVOL_NOLI =U2,  
                                INST =4.),  
                  INCREMENT =_F( LIST_INST =LIST),  
                  ARCHIVAGE =_F( LIST_INST =LIST,  
                                DETR_NUME_SUIV ='OUI')) ,
```

Le résultat de l'archivage pour le 1er U2 est le suivant :

numéro d'archivage	: 1	2	3	4	5
instants correspondants	: 2.	4.	6.	8.	10.

Le résultat final de l'archivage pour U2 est le suivant (par défaut nuinit = 3):

numéro d'archivage	: 1	2	3	4	5
instants correspondants	: 2.	4.	6.	8.	10.

## C - Exemple avec NUME\_INIT

```
LIST = DEFI_LIST_REEL( DEBUT =0.,  
                      INTERVALLE =_F(JUSQU'A =10., NOMBRE =5)),  
  
U2 = STAT_NON_LINE( INCREMENT =_F (LIST_INST =LIST)) ,  
  
U2 = STAT_NON_LINE( reuse=U2,  
                  ETAT_INIT =_F( EVOL_NOLI =U2,  
                                INST =4.),  
                  INCREMENT =_F( LIST_INST =LIST),  
                  ARCHIVAGE =_F( LIST_INST =LIST,  
                                NUME_INIT =2 ,  
                                DETR_NUME_SUIV ='OUI')) ,
```

Le résultat de l'archivage pour le 1er U2 est le suivant :

numéro d'archivage	: 1	2	3	4	5
instants correspondants	: 2.	4.	6.	8.	10.

Le résultat final de l'archivage pour U2 est le suivant :

numéro d'archivage	: 1	2	3	4
instants correspondants	: 2.	6.	8.	10.

### 3.15.4 Opérande CHAM\_EXCLU

```
◇ CHAM_EXCLU = | 'DEPL'  
                | 'SIEF_ELGA'  
                | 'VARI_ELGA'  
                | 'VARI_NON_LOCAL'  
                | 'LANL_ELGA'
```

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

### 3.16 Opérande OBSERVATION

La syntaxe de ce mot clé commun à la commande *DYNA\_NON\_LINE* est décrite dans le document [U4.53.01].

### 3.17 Opérande SOLV\_NON\_LOCAL

La syntaxe de ce mot clé est identique au mot clé *SOLVEUR* décrit dans le document [U4.50.01]. A utiliser pour un modèle non local.

### 3.18 Opérande LAGR\_NON\_LOCAL

L'intégration de lois de comportement non locales impose la résolution d'un problème global (sur toute la structure) : la minimisation d'une fonctionnelle énergie (l'expression du lagrangien augmenté) par rapport à une variable nodale scalaire.

La résolution de ce problème s'effectue au moyen d'un algorithme newton primal et BFGS dual combiné, qui consiste en deux phases :

- Résolution du problème primal :
  - Minimisation par rapport à la variable interne non locale et son gradient (*cham\_elem*)
  - Minimisation par rapport à la variable interne aux nœuds (*cham\_no*)
  - Test de convergence primal : la plus grande composante du résidu assemblé
- Résolution du problème dual : (Maximisation par rapport aux multiplicateurs de Lagrange)
  - Calcul d'une direction de descente BFGS
  - Recherche linéaire par méthode de Wolfe
  - Test de convergence dual : la plus grande composante du gradient
  - Réactualisation des multiplicateurs de Lagrange

◇ **ITER\_PRIM\_MAXI : iterprimmax (10 par défaut)**  
Nombre d'itérations maximales pour la résolution du problème primal.

◆ **RESI\_PRIM\_ABSO : resiprimab**  
Précision pour le test de convergence pour le problème primal.

◇ **ITER\_DUAL\_MAXI : iterdmax (50 par défaut)**  
Nombre d'itérations maximales pour la résolution du problème dual.

◆ **RESI\_DUAL\_ABSO : residabso**  
Précision pour le test de convergence pour le problème dual.

◇ **R : rho (1000 par défaut)**

Coefficient de pénalisation du lagrangien augmenté.

**Remarque :**

*Comme la précision du problème dual dépend fortement de celle du problème primal, on conseille de choisir une meilleure précision pour le problème primal, par exemple 100 ou 1000 fois plus que pour le problème dual.*

## 3.19 Opérande INFO

◇ **INFO : inf**

Permet d'effectuer dans le fichier message diverses impressions intermédiaires en présence de contact unilatéral traité par la méthode des contraintes actives.

inf = 1 impression de la liste des nœuds en contact après convergence à chaque itération de Newton.  
= 2 idem 1 plus impression des associations/dissociations de nœuds entre itérations de la méthode des contraintes actives.

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.

## 3.20 Opérande TITRE

◇ **TITRE : tx**

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

Page laissée intentionnellement blanche.