
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) soit au contact-frottement. 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

Table des Matières

1 But.....	1
2 Syntaxe.....	4
3 Opérandes.....	8
3.1 Opérandes MODELE / CHAM_MATER / CARA_ELEM	8
3.2 Mot clé EXCIT.....	8
3.2.1 Opérandes CHARGE	8
3.2.2 Opérande FONC_MULT.....	9
3.2.3 Opérande TYPE_CHARGE.....	9
3.3 Mot-clé SOUS_STRUC.....	10
3.4 Mots-clés COMP_INCR et COMP_ELAS.....	10
3.5 Mot clé ETAT_INIT.....	10
3.5.1 Opérandes SIGM / VARI / DEPL.....	10
3.5.2 Opérandes EVOL_NOLI.....	11
3.5.3 Opérande NUME_ORDRE / INST / NUME_DIDI.....	11
3.5.4 Opérande INST_ETAT_INIT.....	11
3.5.5 Opérande PRECISION / CRITERE.....	12
3.6 Mot clé INCREMENT.....	14
3.6.1 Opérande LIST_INST	14
3.6.2 Opérandes NUME_INST_INIT/INST_INIT/NUME_INST_FIN/INST_FIN.....	14
3.6.3 Opérande PRECISION.....	15
3.6.4 Opérande ERRE_TEMPS.....	16
3.7 Opérande METHODE.....	16
3.8 Mot clé NEWTON.....	16
3.8.1 Opérande PREDICTION.....	16
3.8.2 Opérande MATRICE.....	17
3.8.3 Opérande EVOL_NOLI.....	18
3.9 Mot clé IMPL_EX.....	18
3.9.1 Opérande MATRICE.....	18
3.10 Mot clé RECH_LINEAIRE.....	18
3.10.1 Opérande METHODE.....	18
3.10.2 Opérande RESI_LINE_RELA / ITER_LINE_MAXI.....	18
3.10.3 Opérande PAS_MINI_CRIT / ITER_LINE_CRIT.....	19
3.10.4 Opérandes RHO_MIN/RHO_MAX/RHO_EXCL.....	19
3.11 Mot clé PILOTAGE.....	19
3.11.1 Opérande TYPE.....	20
3.11.2 Opérandes NOEUD / GROUP_NO	21
3.11.3 Opérandes TOUT / MAILLE / GROUP_MA.....	21
3.11.4 Opérande NOM_CMP.....	21

3.11.5 Opérande COEF_MULT.....	22
3.11.6 Opérande ETA_PILO_R_MAX / ETA_PILO_R_MIN	22
3.11.7 Opérande ETA_PILO_MAX/ETA_PILO_MIN	22
3.11.8 Opérande PROJ_BORNES.....	23
3.11.9 Opérande SELECTION.....	23
3.12 Mot clé SOLVEUR.....	23
3.13 Mot clé CONVERGENCE.....	23
3.13.1 Opérande RESI_GLOB_RELA / RESI_GLOB_MAXI	23
3.13.2 Opérande RESI_COMP_RELA	24
3.13.3 Opérande RESI_REFE_RELA	25
3.13.4 Opérande ITER_GLOB_MAXI	25
3.13.5 Opérande ITER_GLOB_ELAS	25
3.13.6 Opérandes TYPE/PLATEAU_ITER/PLATEAU_RELA.....	25
3.13.7 Opérande ARRET.....	26
3.14 Mot-clé CRIT_FLAMB.....	26
3.15 Mot-clé SENSIBILITE.....	27
3.16 Mot clé ARCHIVAGE.....	27
3.16.1 Opérande LIST_INST / INST / PAS_ARCH.....	27
3.16.2 Opérande PRECISION.....	27
3.16.3 Opérande ARCH_ETAT_INIT / NUME_INIT / DETR_NUME_SUIV.....	27
3.16.4 Opérande CHAM_EXCLU.....	29
3.17 Mot clé AFFICHAGE.....	29
3.17.1 Opérande UNITE.....	29
3.17.2 Opérande NOM_COLONNE.....	29
3.17.3 Opérande INFO_RESIDU.....	31
3.17.4 Opérandes LONG_R, PREC_R et LONG_I.....	31
3.18 Opérande OBSERVATION.....	32
3.19 Opérande INFO.....	33
3.20 Opérande TITRE.....	33

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'
                        ),

◆  SOUS_STRUC          =  _F(
  ◆  CAS_CHARGE         =  chi,                    [char_meca]
  ◆  /TOUT              =  'OUI',                  [DEFAULT]
    /SUPER_MAILLE       =  lma,                    [l_maille]
    FONC_MULT           =  fmult,
                        ),

◆  |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]
    |VARI               =  vain,                  [cham_elem_VARI_R]
    |DEPL               =  depl,                  [cham_no_DEPL_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]
                        ),

◆  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-3,               [DEFAULT]
                        /prec,                    [R]
    /ERRE_TEMPS         =  /'NON'                  [DEFAULT]
                        =  /'OUI'
                        ),

◆  METHODE             =  /'NEWTON',               [DEFAULT]
                        /'IMPL_EX',

◆  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]
REAC_ITER_ELAS      =  /0,                    [DEFAULT]
                      /it,                    [I]
PAS_MINI_ELAS       =  /0,                    [DEFAULT]
                      /pasmini,              [R]
                      ),

♦ IMPL_EX            =  _F(
  MATRICE            =  /'TANGENTE',          [DEFAULT]
  REAC_ITER          =  1,                    [DEFAULT]
                      ),

♦ RECH_LINEAIRE      =  _F(
  METHODE            =  /'CORDE'              [DEFAULT]
                      /'MIXTE'
  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]
                      /rexcl                  [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          =  rno,                  [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',
    ),

♦ 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_COMP_RELA = rescmp, [R]
    | RESI_REFE_RELA = resref, [R]
    SIGM_REFE = sigref, [R]
    EPSI_REFE = sigref, [R]
    DEPL_REFE = depref, [R]
    FORC_REFE = forref, [R]
    VARI_REFE = varref, [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]
    TYPE = /'PIC' [DEFAULT]
    /'PLATEAU'
    PLATEAU_ITER = /3 [DEFAULT]
    /plaite [I]
    PLATEAU_RELA = /1.E-3 [DEFAULT]
    /plarel [R]
    ARRET = /'OUI', [DEFAULT]
    /'NON',
),

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

♦ SENSIBILITE = _F(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',
    | 'SIEF_ELGA',
    | 'VARI_ELGA',
),

♦ AFFICHAGE = _F(
    UNITE = /unite [I]
    LONG_R = /12 [DEFAULT]
```

```

PREC_R      = /long_r      [I]
              /5            [DEFAULT]
              /prec_r      [I]
LONG_I      = /6            [DEFAULT]
              /long_i      [I]
NOM_COLONN  = | 'STANDARD', [DEFAULT]
              | 'MINIMUM',
              | 'ITER_NEWT',
              | 'INCR_TPS',
              | 'RESI_RELA',
              | 'RELA_NOEU',
              | 'RESI_MAXI',
              | 'MAXI_NOEU',
              | 'RESI_REFE',
              | 'RESI_COMP',
              | 'COMP_NOEU',
              | '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'
            ),

♦ OBSERVATION = _F(voir le document [U4.53.01]),

♦ 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 :

- 1) le nom du modèle (mo) dont les éléments font l'objet du calcul mécanique,
- 2) le nom du champ de matériau (chmat) affecté sur le maillage. Attention, toutes les mailles principales du modèle doivent être associées à un matériau (sinon erreur fatale avec message peu explicite),
- 3) 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 (solllicitations et conditions aux limites), et éventuellement un coefficient multiplicateur et/ou un type de charge.

3.2.1 Opérandes CHARGE

- ◆ **CHARGE** : ch_i

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

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 2π (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 2π 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 ».

3.2.2 Opérateur 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 :

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

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

Par défaut : $f_i = 1$.

3.2.3 Opérateur 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 [§19]. Les charges pilotables doivent être issues d'AFFE_CHAR_MECA ou d'AFFE_CHAR_MECA_F (si ce n'est pas une fonction dépendant du temps) 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 Mot-clé SOUS_STRUC

Pour plus de précision concernant l'utilisation de sous-structures (élastiques linéaires) dans une structure non linéaire, on se reportera à la documentation [U2.07.02] et le cas-test ssnv193a.

◆ SOUS_STRUC

Ce mot clé facteur permet de préciser quels sont les chargements à utiliser pour les sous-structures. 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.

◆ CAS_CHARGE = *nocas*

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

◆ /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.

FONC_MULT : *f*

f est la fonction du temps multiplicatrice du chargement précisé dans *CAS_CHARGE*.

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].

3.5 Mot clé ETAT_INIT

◆ ETAT_INIT :

État 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.

Remarques :

- 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.
- Dans le cas où l'on utilise la méthode continue du contact, la reprise de calcul peut donner lieu à des difficultés de convergence du fait de « l'oubli » de l'état de contact précédent.

3.5.1 Opérandes SIGM / VARI / DEPL

◆ / | SIGM = *sig* | VARI = *vain* | DEPL = *depl*

Respectivement, champs de contraintes aux points de Gauss, de variables internes aux points de Gauss et de déplacements aux nœuds 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 CREA_CHAMP, ou bien avoir été lus dans un fichier au format I-DEAS ou MED par la commande LIRE_RESU.

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. [§9]). 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 :

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

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)) ,
```

Premier STAT_NON_LINE : effectue le calcul pour les instants 1, 2, 3 et 4s .

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

Premier STAT_NON_LINE : effectue le calcul des instants 1 à 10s .

Second STAT_NON_LINE : effectue le calcul des instants 21 à 30s, l'état initial correspondant à l'instant $t=10s$ du premier STAT_NON_LINE (par défaut INST=10.). Cet état initial correspond pour ce second 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.)) ,
```

Premier STAT_NON_LINE : effectue le calcul des instants 1 à 10s .

Second STAT_NON_LINE : effectue le calcul des instants 1 à 10s, l'état initial correspondant à l'instant $t=10s$ du premier STAT_NON_LINE (par défaut INST=10.). Cet état initial correspond pour ce second STAT_NON_LINE à l'instant $t=0s$. (INST_ETAT_INIT= 0.).

3.5.5 Opérande PRECISION / CRITERE

PRECISION : **prec** Cf. [U4.71.00] pour la syntaxe détaillée

Ce paramètre sert à repérer le bon numéro d'ordre (NUME_ORDRE) quand l'utilisateur renseigne l'instant (INST). En effet, les instants dans STAT_NON_LINE sont repérés par un numéro d'ordre (un

entier). Si l'utilisateur veut utiliser un instant (un réel) et non un numéro d'ordre pour INST, l'opérande précision permet de sélectionner ce numéro d'ordre.

Exemple:

NUME	1	2	3	4	5	6	7
INST	0.0010	0.0020	0.0030	0.0040	0.0050	0.0060	0.0070

Si l'utilisateur veut sélectionner l'instant correspondant à NUME=4, il lui suffit de dire INST=0,004. Par contre, pour le deuxième exemple:

NUME	1	2	3	4	5	6	7
INST	0.10000001	0.10000002	0.10000003	0.10000004	0.10000005	0.10000006	0.10000007

Si l'utilisateur veut sélectionner l'instant correspondant à NUME=4, il ne lui suffit pas de dire INST=0,10000004, car l'écart relatif entre les instants vaut $\frac{0,10000005 - 0,10000004}{0,10000004} = 1E-7$

qui est supérieur à la valeur de précision par défaut (1E-6). On ne pourra donc pas distinguer NUME=3, 4 et 5 (le code s'arrête alors en erreur fatale). Il suffit alors de changer le paramètre PRECISION pour pouvoir sélectionner l'instant (dans l'exemple, PRECISION=1E-8 conviendra).

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'indiquer les incréments de charge et de paramétrer l'évolution d'un éventuel champ de température.

3.6.1 Opérande LIST_INST

◆ LIST_INST : `litps`

Les instants de calcul sont ceux définis dans le concept `litps` par l'opérateur `DEFI_LIST_REEL` [U4.34.01].

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'instant `litps` (`NUME_INST_INIT`). Pour pouvoir accéder par valeur, il est nécessaire que la liste soit ordonnée.

En l'absence des mots clés `INST_INIT` ou `NUME_INST_INIT`, le défaut est calculé de la manière suivante :

- 1) 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,
- 2) 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'instant `litps` (`NUME_INST_INIT=0`).
- 3) En cas d'archivage (voir mot-clef `ARCHIVAGE`), l'instant initial en poursuite est le dernier pas archivé et non celui défini dans `INST_INIT`.

```
/ 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 :

- Si le re-découpage automatique du pas de temps est activé, `NUME_INST_FIN` n'en tient pas compte et travaille toujours sur la liste d'instant initial.

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)) ,
```

Premier STAT_NON_LINE : effectue le calcul pour les instants 1, 2, 3 et 4s.

Second 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)) ,
```

Premier STAT_NON_LINE : effectue le calcul des instants 1 à 4s.

Second 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] pour la syntaxe détaillée

Ce paramètre sert à repérer le bon numéro d'ordre (NUME_INST_FIN/NUME_INST_INIT) quand l'utilisateur renseigne l'instant (INST_FIN/INST_INIT). En effet, les instants dans STAT_NON_LINE sont repérés par un numéro d'ordre (un entier). Si l'utilisateur veut utiliser un instant (un réel) et non un numéro d'ordre pour (NUME_INST_*), l'opérande précision permet de sélectionner ce numéro d'ordre. Exemple:

NUME	1	2	3	4	5	6	7
INST	0.0010	0.0020	0.0030	0.0040	0.0050	0.0060	0.0070

Si l'utilisateur veut sélectionner l'instant correspondant à NUME=4, il lui suffit de dire INST=0,004. Par contre, pour le deuxième exemple:

NUME	1	2	3	4	5	6	7
INST	0.10000001	0.10000002	0.10000003	0.10000004	0.10000005	0.10000006	0.10000007

Si l'utilisateur veut sélectionner l'instant correspondant à NUME=4, il ne lui suffit pas de dire INST=0,10000004, car l'écart relatif entre les instants vaut $\frac{0,10000005-0,10000004}{0,10000004} = 1E-7$

qui est supérieur à la valeur de précision par défaut (1E-6). On ne pourra donc pas distinguer NUME=3, 4 et 5 (le code s'arrête alors en erreur fatale). Il suffit alors de changer le paramètre PRECISION pour pouvoir sélectionner l'instant (dans l'exemple, PRECISION=1E-8 conviendra).

3.6.4 Opérateur ERRE_TEMP

```
ERRE_TEMP = / 'NON' [DEFAULT]
            / 'OUI'
```

Cet opérateur permet d'activer le calcul des indicateurs d'erreur temporel pour les modélisations HM instationnaires. Voir [R4.10.05].

3.7 Opérateur METHODE

```
◇ METHODE = / 'NEWTON'
            / 'IMPL_EX'
```

Permet de choisir la méthode de résolution du problème incrémental non linéaire.

```
/ 'NEWTON'
```

On utilise l'algorithme de Newton-Raphson pour résoudre le problème.

```
/ 'IMPL_EX'
```

On utilise l'algorithme IMPL-EX pour résoudre le problème.

3.8 Mot clé NEWTON

NEWTON :

Précise les caractéristiques de la méthode de résolution du problème incrémental non linéaire (méthode de Newton-Raphson)

3.8.1 Opérateur 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) 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) qui est utilisée si on a choisi MATRICE='ELASTIQUE'.

```
/ 'TANGENTE'
```

On utilise la matrice tangente du problème en vitesse (option RIGI_MECA_TANG).

```
/ 'ELASTIQUE'
```

On utilise la matrice élastique (option RIGI_MECA).

```
/ '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 (§11).

Le déplacement est projeté sur l'ensemble des champs cinématiquement admissible, comme pour la méthode EXTRAPOLE.

Remarque:

Les méthodes 'EXTRAPOLE' et 'DEPL_CALCULE' procèdent à une projection de la solution sur l'ensemble des champs cinématiquement admissibles. On se sert pour cela des conditions aux limites de Dirichlet donné dans le mot-clé EXCIT. Dans ce cas, il n'est pas possible d'utiliser des chargements de Dirichlet de type « cinématique » (opérande AFFE_CHAR_CINE) mais uniquement des chargements de Dirichlet par dualisation (opérande AFFE_CHAR_MECA). Une alarme prévient l'utilisateur dans le cas où Code_Aster n'aurait pas trouvé de chargements de Dirichlet dualisés. Le risque dans ce cas étant que le champ de déplacement ne soit pas cinématiquement admissible.

Utilité :

- 1) 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 premier calcul 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.
- 2) 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 adéquat 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.

3.8.2 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]. La matrice tangente de prédiction est réévaluée tous les mf incréments de temps (mf positif ou nul) et la matrice tangente cohérente (option FULL_MECA) est réévaluée 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 [R]
REAC_ITER_ELAS = / 0 [DEFAULT]
/ it [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.

On peut définir une fréquence de réactualisation de la matrice de décharge avec le mot-clé REAC_ITER_ELAS (analogue de REAC_ITER). Si la matrice de décharge ne dépend pas de l'état de déformation (ce qui est le cas pour les matériaux plastiques mais pas pour les modèles d'endommagement), prendre REAC_ITER_ELAS = 0 (puisque'elle sera la même au cours des itérations).

Utilité :

Cette option peut être utile lorsque le redécoupage automatique du pas de temps 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 des relations de comportement VMIS_ISOT_TRAC, ECMI_ISOT_TRAC, VISC_ISOT_TRAC [U4.51.11]).

3.8.3 Opérande EVOL_NOLI

◇ EVOL_NOLI : evol_noli

Nom du concept de type evol_noli qui servira dans la prédiction par DEPL_CALCULE.

3.9 Mot clé IMPL_EX

IMPL_EX :

Précise les caractéristiques de la méthode de résolution du problème incrémental non linéaire (méthode IMPL-EX)

3.9.1 Opérande MATRICE

◇ MATRICE =
/ 'TANGENTE'
REAC_ITER = 1 [DEFAULT]

La matrice utilisée pour les itérations globales de la méthode est la matrice tangente [R5.03.01]. La matrice tangente de prédiction est réévaluée à tous les incréments de temps

3.10 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 et en présence de contact.

3.10.1 Opérande METHODE

METHODE = / 'CORDE' [DEFAULT]
/ 'MIXTE'

Permet de choisir la méthode de recherche linéaire, c'est-à-dire l'algorithme de recherche du zéro de la fonctionnelle (voir doc [R5.03.01]). La méthode CORDE (par défaut) est la méthode la plus simple, c'est une méthode sécante unidimensionnelle.

La méthode MIXTE est plus élaborée et utilise une méthode sécante avec des bornes variables. Elle est plus efficace lorsque la fonctionnelle n'est pas strictement concave (problèmes avec endommagement ou THM par exemple).

3.10.2 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. Vous ne pouvez mettre plus de 999 itérations de recherche linéaire.

3.10.3 Opérande PAS_MINI_CRIT / ITER_LINE_CRIT

PAS_MINI_CRIT	=	/	0.	[DEFAULT]
		/	<code>pmicri</code>	[R]
ITER_LINE_CRIT	=	/	20	[DEFAULT]
		/	<code>itelic</code>	[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_CRIT`).

3.10.4 Opérands RHO_MIN/RHO_MAX/RHO_EXCL

RHO_MIN	=	/	1.E-2	[DEFAULT]
		/	<code>rmin</code>	[R]
RHO_MAX	=	/	1.E+1	[DEFAULT]
		/	<code>rmax</code>	[R]
RHO_EXCL	=	/	9.E-3	[DEFAULT]
		/	<code>rexc</code>	[R]

Ces mots-clés fixent l'intervalle I dans lequel on calcule le coefficient `RHO` de la recherche linéaire, sous la forme : $I = [rmin, rmax] - [-rexc, rexc]$ [R5.03.01].

3.11 Mot clé PILOTAGE

♦ PILOTAGE :

Lorsque l'intensité η d'une partie du chargement n'est pas connue a priori (chargement dit de référence défini dans `AFFE_CHAR_MECA` ou `AFFE_CHAR_MECA_F` avec charge de type `FIXE_PILO`), 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_PILO`, on ne peut pas utiliser pour le chargement de référence le mot clé `FONC_MULT`.

Attention :

Lorsque le chargement de référence est défini par `AFFE_CHAR_MECA_F`, ce chargement peut être fonction des variables d'espace mais pas du temps. De même, les changements issus de variables de commande (comme la température) qui dépendent du temps ne sont pas utilisables avec le pilotage.

Attention :

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

3.11.1 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 η du chargement de référence qui permettra de satisfaire la relation incrémentale suivante :

$$cmult . \Delta u_i(no) = \Delta t$$

/ 'LONG_ARC'

Permet de piloter l'intensité η 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 :

$$cmult . \|\Delta u\| = \Delta t \text{ avec } \|(\Delta u)\| = \left(\sqrt{\sum_n \sum_c (\Delta u_{n,c}^2)} \right)$$

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_PILO', 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 autre mot clé 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 :

$$cmult . max_{\text{point de Gauss}} \left(\frac{\dot{\epsilon}}{\|\dot{\epsilon}\|} . \Delta \epsilon \right) = \Delta t$$

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 $f_{\text{pred-elas}}$ d'une quantité $\frac{\Delta t}{\text{cmult}}$. On vérifie la relation :

$$\text{cmult} \cdot \max_{\text{point de Gauss}} (f_{\text{pred-elas}}) = \Delta t$$

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

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 et ENDO_ORTH_BETON (avec la version locale et la version non locale), BARENBLATT et BETON_DOUBLE_DP.

La fixation du paramètre cmult est difficile à définir du premier coup parce que la notion de sortie de critère $\frac{\Delta t}{\text{cmult}}$ n'est pas intuitive et varie selon les lois de comportement. Pour les lois ENDO_FRAGILE et ENDO_ISOT_BETON, une version différente de la définition de $\frac{\Delta t}{\text{cmult}}$ est utilisée (voir [R7.01.04]).

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 ε^- 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 ε^- . 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 ε^- 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.

3.11.2 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).

3.11.3 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.

3.11.4 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'.

3.11.5 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 ($P = \eta * \text{valeur de référence } P_x$) sur le groupe de maille A . Pour trouver la charge limite P_{limite} , on va piloter le déplacement du nœud NOI . 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 η sera affichée à chaque instant du calcul. Pour connaître la charge limite, il suffit de faire $P_{limite} = \eta * P_x$. (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.

Attention à la signification de **COEF_MULT** pour le pilotage de type **PRED_ELAS**.

3.11.6 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 $[etarmin, etarmax]$, toutes les valeurs de pilotage en dehors de cet intervalle sont ignorées. 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 imposée à 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.

3.11.7 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.

3.11.8 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 $\text{eta} > \text{etamax}$ alors $\text{eta} = \text{etamax}$; si $\text{eta} < \text{etamin}$ alors $\text{eta} = \text{etamin}$), 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.

3.11.9 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.

Remarque :

| Si on fait une reprise de calcul (reuse) avec le mot-clé SELECTION='ANGL_INCR_DEPL', il est important de garder à l'esprit que ce critère nécessite les DEUX pas de temps précédents. Il faudra donc bien prendre soin d'archiver correctement les résultats du précédent calcul au risque d'obtenir des résultats faux. Une alarme avertit l'utilisateur.

3.12 Mot clé SOLVEUR

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

3.13 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.

3.13.1 Opérande RESI_GLOB_RELA / RESI_GLOB_MAXI

◇ | RESI_GLOB_RELA = resrel

L'algorithme continue les itérations globales tant que :

$$\max_{i=1,\dots,nbddl} |\mathbf{F}_i^n| > \text{resrel} \cdot \max |\mathbf{L}|$$

où \mathbf{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 :

$$\max_{i=1,\dots,nbddl} |\mathbf{F}_i^n| > \text{resmax}$$

où \mathbf{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.

Remarque:

Si les conditions limites de Dirichlet sont imposés par `AFFE_CHAR_CINE` (élimination) et non par `AFFE_CHAR_MECA` (dualisation), les degrés de libertés portant ces conditons sont ignorées lors de l'évaluation du résidu d'équilibre. Ce qui ne provoque pas de résultats faux mais lorsque le chargement devient nul, 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`.

3.13.2 Opérande `RESI_COMP_RELA`

◇ | `RESI_COMP_RELA = rescmp`

Cet opérande conduit à estimer la convergence de l'algorithme de Newton en raisonnant par composant. Pour cela, on distingue dans le résidu les constituants correspondant chaque `cmp` (par exemple en THM, $\text{cmp} = [DX, DY, DZ, PRE1, PRE2, TEMP]$). On norme ensuite ces sous vecteurs par la force interne correspondant au composant. Ce choix n'a d'intérêt que pour des problèmes de type évolutif (THM) où résident de forts contrastes.

Ainsi, l'algorithme continue les itérations globales tant que :

$$\max_{c=1,\dots,nbcmp} \left(\frac{\max_{i=1,\dots,nbddl} |\mathbf{F}_i^{c,n}|}{\max_{i=1,\dots,nbddl} |\mathbf{L}_i^{\text{int},c,n}|} \right) > \text{rescmp}.$$

où $\mathbf{F}^{c,n}$ est la partie du résidu \mathbf{F}^n correspondant au composant c et $\mathbf{L}^{\text{int},c,n}$ le vecteur des forces internes au temps n correspondant à ce même composant c (Cf. [R5.03.01] pour plus de détails).

Les forces internes au temps n sont calculées en début de pas de temps à partir du résultat issu du pas de temps précédent. Pour le premiers pas de temps, on passe automatiquement à un critère relatif de type `RESI_GLOB_RELA`, voir à un critère absolu pour les cas où le chargement est nul.

3.13.3 Opérande `RESI_REFE_RELA`

	<code>RESI_REFE_RELA</code>	=	<code>resref</code> ,	[R]
	<code>SIGM_REFE</code>	=	<code>sigref</code> ,	[R]
	<code>FORC_REFE</code>	=	<code>forref</code> ,	[R]
	<code>VARI_REFE</code>	=	<code>varref</code> ,	[R]
	<code>EPSI_REFE</code>	=	<code>epsref</code> ,	[R]


```

FLUX_THER_REFE = fthref, [R]
FLUX_HYD1_REFE = fh1ref, [R]
FLUX_HYD2_REFE = fh2ref, [R]
DEPL_REFE      = depref, [R]

```

Cet opérateur conduit à estimer la convergence de l'algorithme de Newton de la manière suivante (Cf. [R5.03.01] pour plus de détails). A partir d'une référence, qui peut être:

- une contrainte `sigref`
- une déformation `epsref`
- une variable interne `varref` si l'on utilise des lois non locales à gradient de déformation
- un flux thermique `fthref` dans un cas THM
- deux flux hydriques `fh1ref` et `fh2ref` dans un cas HHM
- un déplacement `depref` si on utilise des éléments de joint avec un comportement de type CZM
- une force ou un moment généralisé `forref` si on utilise des éléments de structure (discrets, barres, poutres ou cables)

On calcule une référence de résidu F^{ref} (un vecteur de même longueur que le vecteur résidu). La convergence sera réalisée si et seulement si :

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

3.13.4 Opérateur 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 sauf dans le cas du redécoupage du pas de temps par la méthode 'EXTRAPOLE'.

3.13.5 Opérateur 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 §16) 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.

3.13.6 Opérateurs TYPE/PLATEAU_ITER/PLATEAU_RELA

```

TYPE           = / 'PIC' [DEFAULT]
                / 'PLATEAU'
PLATEAU_ITER   = / 3 [DEFAULT]
                / plaite [I]
PLATEAU_RELA   = / 1E-3 [DEFAULT]
                / plarel [R]

```

Cet opérateur permet de contrôler le type d'opérateur de convergence à appliquer pour les mots-clés RESI_*. Par défaut, le mode 'PIC' assure qu'il y a convergence dès lors que la valeur seuil donnée par RESI_* est atteinte. Le mode 'PLATEAU' est plus sévère et exige que le critère de convergence soit stable pendant `plaite` itérations autour dans un tunnel de largeur `plarel` autour de la valeur de référence donnée par RESI_*. Ce mode est utile dans les modélisations THM pour lesquelles le critère de Newton est parfois pas assez sévère.

3.13.7 Opérateur 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.

3.14 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, 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 l'objet résultat, 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 §26). 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.

L'analyse de stabilité linéaire ne permettant pas de tenir compte de l'aspect suiveur de certaines forces, il faut alors utiliser **CRIT_FLAMB**.

La documentation [U2.08.04] présentent les différentes approches pour les analyses de stabilité dans Code_Aster, donc **CRIT_FLAMB**.

3.15 Mot-clé SENSIBILITE

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

3.16 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. L'archivage permet de réduire sensiblement la taille des bases en sélectionnant les instants sauvegardés.

Remarques :

| *En présence de contact, on ne peut pas archiver plus de 99 999 instants de calculs.*

3.16.1 Opérateur 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`) 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 :

- 1) *le dernier pas de calcul est toujours stocké pour pouvoir effectuer une reprise,*
- 2) *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*

3.16.2 Opérateur PRECISION

```
PRECISION = /1.e-6 [DEFAULT]  
           /prec  
Cf. [U4.71.00]
```

3.16.3 Opérateur 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^{er} 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 :

- 1) si l'état initial n'est pas fixé par le concept calculé, il s'agit du dernier numéro d'ordre +1 (exemple A),
- 2) 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 premier 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 (les numéros d'ordre 3, 4 et 5 sont recalculés et archivés):

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 premier 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.

3.16.4 Opérande CHAM_EXCLU

```
CHAM_EXCLU = | 'DEPL'  
              | 'SIEF_ELGA'  
              | 'VARI_ELGA'
```

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

3.17 Mot clé AFFICHAGE

Ce mot-clef facteur permet de personnaliser l'affichage du tableau de convergence dans STAT_NON_LINE.

AFFICHAGE

Si ce mot-clef n'est pas renseigné, le tableau est affiché en mode 'STANDARD' et avec INFO_RESIDU='NON'.

Chaque occurrence d'AFFICHAGE concerne l'affichage d'une colonne et son format. L'ordre des colonnes donné par la succession des NOM_COLONNE est respecté.

3.17.1 Opérande UNITE

UNITE =unit

Le tableau de convergence sera dupliqué dans le fichier d'unité *unit*.

Remarque :

L'unité peut être répétée à chaque occurrence du mot-clef facteur mais seul la première est prise en compte (avec affichage d'une alarme).

3.17.2 Opérande NOM_COLONNE

NOM_COLONNE =	'STANDARD',	[DEFAULT]
	'MINIMUM',	
	'ITER_NEWT',	
	'INCR_TPS',	
	'RESI_RELA',	
	'REL_NOEU',	
	'RESI_MAXI',	
	'MAXI_NOEU',	
	'RESI_REFE',	
	'REFE_NOEU',	
	'RESI_COMP',	
	'COMP_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',	

Type de la colonne à afficher (chaque valeur correspond à une colonne affichée):

ITER_NEWT

numéro de l'itération de Newton en cours. La colonne est marquée par un « X » tant qu'il n'y a pas eu convergence sur tous les critères.

INCR_TPS

instant de calcul courant.

RESI_RELA et REL_NOEU

valeur de RESI_GLOB_RELA et affichage du nœud où il est maximum. La colonne est marquée par un « X » tant que le résidu est plus grand que celui spécifié par l'utilisateur (opérande RESI_GLOB_RELA) ou tant que le critère n'est pas dans le plateau.

RESI_MAXI et MAXI_NOEU

valeur de RESI_GLOB_MAXI et affichage du nœud où il est maximum. La colonne est marquée par un « X » tant que le résidu est plus grand que celui spécifié par l'utilisateur (opérande RESI_GLOB_MAXI) ou tant que le critère n'est pas dans le plateau.

RESI_REFE et REFE_NOEU

valeur de RESI_REFE_RELA et affichage du nœud où il est maximum. La colonne est marquée par un « X » tant que le résidu est plus grand que celui spécifié par l'utilisateur (opérande RESI_REFE_RELA) ou tant que le critère n'est pas dans le plateau.

RESI_COMP et COMP_NOEU

valeur de RESI_COMP_RELA et affichage du nœud où il est maximum. La colonne est marquée par un « X » tant que le résidu est plus grand que celui spécifié par l'utilisateur (opérande RESI_COMP_RELA) ou tant que le critère n'est pas dans le plateau.

RELI_ITER et RELI_COEF

nombre d'itérations et coefficient de recherche linéaire.

PILO_PARA

valeur du paramètre de pilotage.

MATR_ASSE

option d'assemblage pour la matrice: ELASTIQUE, TANGENTE, SECANTE. DEPL_CALCULE, EXTRAPOL. S'il n'y a rien, c'est qu'on n'assemble pas la matrice (selon la valeur des mots-clefs REAC_ITER et REAC_INCR).

ITER_DEBO

indique une itération supplémentaire pour satisfaire les conditions de contraintes planes ou unidimensionnelles. ALGO_C_PLAN = 'DEBORST' ou ALGO_1D = 'DEBORST' contraintes planes ou unidimensionnelles (voir COMP_INCR [U4.51.11]).

SUIV_1/SUIV_2/SUIV_3/SUIV_4

indique une occurrence de SUIVI_DDL permettant de suivre une valeur en un point particulier (voir DYNA_NON_LINE).

ITER_FETI

indique le nombre d'itérations du solveur d'interface pour la méthode FETI.

Pour le contact méthodes discrètes :

CTCD_ITER

nombre d'itérations internes de contact/frottement, méthodes discrètes. La colonne est marquée par un « X » si le contact attend un point fixe (METHODE='LAGRANGIEN').

CTCD_GEOM

valeur du déplacement maximum pour la réactualisation géométrique du contact, méthodes discrètes.

CTCD_NOEU

nœud où la valeur du déplacement est maximale lors de la réactualisation géométrique du contact, méthodes discrètes.

Pour le contact méthode continue et contact XFEM :

CTCC_GEOM

numéro de l'itération de contact méthode continue lors de la boucle sur la géométrie. La colonne est marquée par un « X » tant qu'on n'a pas convergé.

CTCC_FROT

numéro de l'itération de contact méthode continue lors de la boucle sur le seuil de frottement. La colonne est marquée par un « X » tant qu'on n'a pas convergé.

CTCC_CONT

numéro de l'itération de contact méthode continue lors de la boucle sur l'état de contact (contraintes actives). La colonne est marquée par un « X » tant que l'on n'a pas convergé.

Types composites (affiche plusieurs colonnes) :

STANDARD

affichage standard (par défaut) du tableau de convergence. Contient :

- 1) Le numéro de l'itération de Newton (ITER_NEWT)
- 2) Toutes les colonnes nécessaires selon les fonctionnalités activés (recherche linéaire, contact, pilotage,...)
- 3) La valeur des résidus (RESI_MAXI et RESI_RELA)

MINIMUM

affichage minimum du tableau de convergence. Contient :

- 1) Le numéro de l'itération de Newton (ITER_NEWT)
- 2) La valeur des résidus (RESI_MAXI et RESI_RELA)

Remarques :

- 1) On ne peut demander plus de seize colonnes (16 colonnes de 16 caractères, soit une largeur totale de 256)
- 2) Les colonnes sont cumulables : on peut demander l'affichage MINIMUM et ajouter une colonne quelconque
- 3) On peut avoir plusieurs fois la même colonne
- 4) Tant que « X » est affiché dans la colonne ITER_NEWT, le calcul n'a pas convergé. Ceci dépend bien sûr de la valeur des résidus mais aussi de la convergence du contact ou de De Borst.
- 5) Pour la méthode de contact continue, les itérations de Newton constitue une boucle interne aux trois autres boucles (CTCC_GEOM, CTCC_FROT et CTCC_CONT). ITER_NEWT n'est donc pas en première position en mode 'STANDARD' et c'est le marquage des colonnes CTCC_* qui joue le rôle de juge de paix final sur la convergence.

3.17.3 Opérande INFO_RESIDU

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

Cet opérande permet d'ajouter une colonne pour chaque résidu évalué (RESI_RELA, RESI_MAXI et RESI_REFE). Cette colonne indiquera le nœud où le résidu est maximum, ce qui peut aider l'utilisateur lorsqu'il y a des difficultés de convergence. Par exemple, pour voir si le matériau a été mal défini avec une valeur incorrecte sur un élément.

Cette option est strictement équivalente à l'ajout des colonnes RELA_NOEU, RELA_MAXI ou RELA_REFE quand on décrit complètement l'affichage du tableau de convergence mais permet d'afficher l'information sur les nœuds lorsque l'on est en mode 'STANDARD' ou 'MINIMUM', sans avoir besoin de décrire toutes les autres colonnes.

3.17.4 Opérandes LONG_R, PREC_R et LONG_I

◇ LONG_R	=	/12	[DEFAULT]
		/long_r	[I]
PREC_R	=	/5	[DEFAULT]
		/prec_r	[I]
LONG_I	=	/6	[DEFAULT]
		/long_i	[I]

Ces opérandes permettent de modifier l'affichage des informations dans le tableau de convergence. Toutes les colonnes ont une largeur fixe de 16 caractères. Quand l'information est un réel, on peut demander un affichage personnalisé : la longueur long_r du réel affiché (maximum 16) et le nombre de chiffres significatifs.

Quand c'est un entier, on peut régler la longueur par long_i. Pour une chaîne de caractères, le format est toujours de 16 caractères.

3.18 Opérande OBSERVATION

La syntaxe de ce mot clé commun à la commande DYNA_NON_LINE est décrite dans le document [U4.53.01].

3.19 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.

3.20 Opérande TITRE

◇ TITRE : tx

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