Opérateur STAT_NON_LINE

1 Drank

Compute the mechanical evolution or coupled thermo-hydro-mechanics, into quasi-static, of a structure into nonlinear.

Nonthe linearity is related either to the behavior of the material (for example plastic), or with the geometry (for example in large displacements) or with contact-friction. To have details on the method of resolution employed, one will refer to documentation of reference [R5.03.01].

The evolution can be studied in several successive works (D-entering concept), either in continuation (the computed last moment is the initial time of following computation), or recovery some on the basis of one former time.

If the time necessary to carry out computation is not sufficient, the program stops, but the already computed results are saved if a data base were defined in the profile of study of the user. Product a data structure of the evol_noli `type`.

Contents

2 Syntaxe

```
statnl [evol_noli] = STAT_NON_LINE
(   reuse           =  statnl,                          [evol_noli]

♦   MODELE=mo                ,                             [model]

♦   CHAM_MATER=chmat             ,                        [cham_mater]

◊   CARA_ELEM=carac              ,                        [cara_elem]

♦   EXCIT=_F               (
    ♦  CHARGE=chi             ,                          [char_meca]
       FONC_MULT=fi           ,
[function/formulate]
       TYPE_CHARGE=        /"FIXE_CSTE"                  [DEFECT]
                           /"FIXE_PILO"
                           /"SUIV"
                           /"DIDI"
                     ),

◊   CONTACT=contact              ,                        [char_contact]

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

♦   |         COMP_INCR=_F  (see the document [U4.51.11]),
    |         COMP_ELAS=_F  (see the 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]
         |STRX=strx                                        [cham_elem_STRX_R]
       /EVOL_NOLI       =evol  ,                         [evol_noli]
       /NUME_ORDRE      =nuini ,                         [I]
       /INST            =instini  ,                      [R]
   ◊PRECISION=                /1.0E-3,                     [DEFAUT]
                           /prec,                        [R]
   ◊CRITERE=               /"RELATIF",                   [DEFECT]
                           /"ABSOLU",
   ◊NUMÉRIQUE_DIDI=nudidi                ,                  [I]
   ◊INST_ETAT_INIT=istetaini                              [R]
                     ),

♦   INCREMENT=_F               (
    ♦ LIST_INST=            /litpsr8,                     [listr8]
                            /litps,                       [list_inst]

       /NUME_INST_INIT     =nuini ,                       [I]
       /INST_INIT          =instini  ,                    [R]
       /NUME_INST_FIN      =nufin ,                       [I]
       /INST_FIN           =INSTFIN  ,                    [R]
        PRECISION=            /1.0E-3,                    [DEFAUT]
```

```
                                /prec,                  [R]

                        ),

      ◊  CRIT_QUALITE        =_F  (
         ◊  /  ERRE_TEMPS_THM=  /"NON"                  [DEFECT]
                                =  /"OUI"
                                  )

      ◊  METHODE=               /"NEWTON",               [DEFECT]
                                /"IMPLEX",
                                /"NEWTON_KRYLOV",

      ◊  NEWTON=_F               (
            PREDICTION=         /"TANGENTE",             [DEFECT]
                                /"ELASTIQUE",
                                /"EXTRAPOLE",
                                /"DEPL_CALCULE",
            EVOL_NOLI=evol_noli          ,              [evol_noli]
            MATRICE=            /"TANGENTE",             [DEFECT]
                                /"ELASTIQUE"
            REAC_INCR=             /1,                      [DEFAUT]
                                /mf,                     [I]
            REAC_ITER=             /0,                      [DEFAUT]
                                /it,                     [I]
            REAC_ITER_ELAS=     /0,                      [DEFAUT]
                                /it,                     [I]
            PAS_MINI_ELAS=         /0,                      [DEFAUT]
                                /pasmini,                [R]
                        ),

      ◊RECHERCHE_LINEAIRE=_F              (
            METHODE=            /"ROPE"                  [DEFECT]
                                /"MIXTE"
                                /"PILOTAGE"
            RESI_LINE_RELA=     /1.E-1,                  [DEFAUT]
                                /reslin,                 [R]
            ITER_LINE_MAXI=     /3                       [DEFAUT]
                                /itelin                 [I]
            RHO_MIN=            /1.E-2                   [DEFAUT]
                                /rmin                   [R]
            RHO_MAX=            /1.E+1                   [DEFAUT]
                                /rmax                   [R]
            RHO_EXCL=           /9.E-3                   [DEFAUT]
                                /rexc                   [R]
                        ),

      ◊  PILOTAGE=_F              (
          ♦  TYPE=              /"DDL_IMPO",
                                /"SAUT_IMPO",
                                /"LONG_ARC",
                                /"SAUT_LONG_ARC"
                                /"ANA_LIM",
                                /"DEFORMATION",
                                /"PRED_ELAS",
            TOUT=               'OUI',                   [DEFAUT]
            /GROUP_MA        =lgrma ,                    [l_gr_maille]
            /MAILLE          =lma   ,                    [l_maille]
            /NOEUD           =no ,                       [node]
            /GROUP_NO        =rno   ,                    [gr_noeud]
```

```
          FISSURE=fiss          ,                          [sd_fiss_xfem],
          NOM_CMP=nomcmp            ,                      [kN]
          /DIRE_PILO      =direpilo ,                      [kN]
          COEF_MULT=             /1. ,                        [DEFAUT]
                                /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"                      [DEFECT]
                                /"NON"
          EVOL_PARA=               /"SANS"                     [DEFECT]
                                /"DECROISSANT"
                                /"CROISSANT"
          SELECTION=               /"NORM_INCR_DEPL",          [DEFECT]
                                /"ANGL_INCR_DEPL",
                                /"RESIDU",
                           ),
```

◊ SOLVEUR=_F (see the document [U4.50.01]),

```
◊ CONVERGENCE=_F             (
          /RESI_GLOB_RELA  =1.E-6 ,                         [DEFAUT]
          /| 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          ,                       [l_R]
          VARI_REFE=varref          ,                       [R]
          FLUX_THER_REFE=sigref     ,                       [R]
          FLUX_HYD1_REFE=sigref     ,                       [R]
          FLUX_HYD2_REFE=sigref     ,                       [R]
          ITER_GLOB_ELAS=      /25,                         [DEFAUT]
                               /maxelas,                    [I]
          ITER_GLOB_MAXI=      /10,                         [DEFAUT]
                               /maglob,                     [I]
          TYPE=                /"PIC"                       [DEFECT]
                               /"PLATEAU"
          PLATEAU_ITER=        /3                           [DEFAUT]
                               /plaite                     [I]
          PLATEAU_RELA=        /1.E-3                       [DEFAUT]
                               /plarel                     [R]
          ARRET=               /"OUI",                      [DEFECT]
                               /"NON",
                           ),

◊ CRIT_STAB=_F               (
          TYPE=                /"BUCKLING"                  [DEFECT]
                           =   /"STABILITY"
           NB_FREQ=            /3,                          [DEFAUT]
                                /nbfreq,                    [I]
          COEF_DIM_ESPACE=     /5,                          [DEFAUT]
                               /coef,                       [I]
          RIGI_GEOM=               /"OUI",                     [DEFECT]
                           =   /"NON",
          MODI_RIGI=               /"NON",                     [DEFECT]
                           =   /"OUI",
```

```
        CHAR_CRIT=              /(- 10.10),                    [DEFAUT]
                              /intcc,
        DDL_EXCLUS=liste1_ddl
        DDL_STAB=liste2_ddl
        /LIST_INST      =liste_r8 ,                          [listr8]
        /INST           =l_r8  ,                             [R]
        /PAS_CALC       =npas  ,                             [I]
        ◊PRECISION=             /1.E-6,                        [DEFAUT]
                              /prec,                         [R]
        ◊   CRITERE=          /"RELATIF",                    [DEFECT]
                              /"ABSOLU",
        SIGNE=                /"POSITIF_NEGATIF",            [DEFECT]
                            =   /"POSITIF",
                            =   /"NEGATIF",
        PREC_INSTAB=          /1.E-6,                        [DEFAUT]
                              /prec_instab,                  [R] ),

◊ENERGIE=_F                       ()

◊ARCHIVAGE=_F                     (
        /LIST_INST      =liste_r8 ,                          [listr8]
        /INST           =l_r8  ,                             [R]
        /PAS_ARCH       =npas  ,                             [I]
        ◊PRECISION=             /1.E-6,                        [DEFAUT]
                              /prec,                         [R]
        ◊CRITERE=             /"RELATIF",                    [DEFECT]
                              /"ABSOLU",
        ◊CHAM_EXCLU=              list_txt,
                              ),

◊  AFFICHAGE=_F                   (
        ◊UNITE=                 /unit                          [I]
        ◊INFO_RESIDU=         /"NON",                        [DEFECT]
                              /"OUI"
        ◊INFO_TEMPS=          /"NON",                        [DEFECT]
                              /"OUI"
                            ),
◊  OBSERVATION              = _F    (
        ◊  TITLE            =titre  ,                  [list_k]
        ♦  NOM_CHAM=            |'DEPL',
                                |'VITE',
                                |'ACCE',
                                |'DEPL_ABSOLU',
                                |'VITE_ABSOLU',
                                |'ACCE_ABSOLU',
                                |'SIEF_ELGA',
                                |'VARI_ELGA',
                                |'FORC_NODA',
                                |'VALE_CONT',
        ◊EVAL_CHAM          =   /"VALE",                     [DEFECT]
                                /"MIN",
                                /"MAX",
                                /"MOY",
                                /"MINI_ABS",
                                /"MAXI_ABS",
        ♦NOM_CMP=lnocmp                  ,                    [l_Kn]
        ◊EVAL_CMP           =   /"VALE",                     [DEFECT]
                                /"FORMULA",
    {If EVAL_CMP='FORMULATE'
          ◊FORMULE          =    form                        [formule_aster]
```

```
        }
      {If CHAM is of type ELGA    (SIEF_ELGA, VARI_ELGA)
            ◊TOUT=              'OUI',                        [DEFAUT]
            ◊  /      GROUP_MA=lgrma   ,                        [l_gr_maille]
            ◊  /MAILLE=lma         ,                          [l_maille]
            ◊  EVAL_ELGA=          /"VALE",                     [DEFECT]
                                   /"MIN",
                                   /"MAX",
         {If EVAL_ELGA ='VALE'
            ♦POINT=pi                      ,                   [I]
            ◊SOUS_POINT=spi                ,                   [I]
         }
      }
      {If CHAM is of type NOEU
         ◊   TOUT=           'OUI',                           [DEFAUT]
         ◊  /      GROUP_MA=lgrma   ,                           [l_gr_maille]
         ◊  /MAILLE=lma         ,                             [l_maille]
         ◊  /NOEUD=no           ,                               [node]
         ◊ /GROUP_NO     =rno   ,                             [gr_noeud]
        }
      ◊ /LIST_INST =linst ,                                 [listr8]
      ◊ /INST         =linst ,                                 [l_R]
      ◊ /PAS_OBSE  =pas      ,                                   [I]
      ◊CRITERE=       /"RELATIF",                           [DEFECT]
                      /"ABSOLU",
         { If   CRITERE=     'RELATIF'
            ◊  PRECISION=       /1.0E-6,                        [DEFAUT]
                              /prec,                          [R]
         }
         { If   CRITERE=     'ABSOLU'
             ♦ PRECISION=prec      ,                            [R]
          }
       ),
◊   SUIVI_DDL       = _F    (
         ◊  TITLE        =titre  ,                        [list_k]
         ♦  NOM_CHAM=        |'DEPL',
                             |'VITE',
                             |'ACCE',
                             |'SIEF_ELGA',
                             |'VARI_ELGA',
                             |'FORC_NODA',
         ◊EVAL_CHAM      =  /"VALE",                        [DEFECT]
                            /"MIN",
                            /"MAX",
                            /"MOY",
                            /"MINI_ABS",
                            /"MAXI_ABS",
         ♦NOM_CMP=lnocmp                ,                     [l_Kn]
         ◊EVAL_CMP       =  /"VALE",                        [DEFECT]
                            /"FORMULA",
      {If EVAL_CMP='FORMULATE'
            ◊FORMULE        =   form                        [formule_aster]
       }
      {If CHAM is of type ELGA    (SIEF_ELGA, VARI_ELGA)
            ◊TOUT=              'OUI',                        [DEFAUT]
            ◊  /      GROUP_MA=lgrma   ,                        [l_gr_maille]
            ◊  /MAILLE=lma         ,                          [l_maille]
            ◊  EVAL_ELGA=          /"VALE",                     [DEFECT]
                                   /"MIN",
                                   /"MAX",
```

```
        {If EVAL_ELGA ='VALE'
           ♦POINT=pi                    ,                    [I]
           ◊SOUS_POINT=spi              ,                    [I]
        }
     }
     {If CHAM is of type NOEU
        ◊    TOUT=           'OUI',                          [DEFAUT]
        ◊   /      GROUP_MA=lgrma   ,
[l_gr_maille]
        ◊   /MAILLE=lma         ,                            [l_maille]
        ◊   /NOEUD=no           ,                               [node]
        ◊  /GROUP_NO     =rno   ,                            [gr_noeud]
      }
     ),
◊INFO=                     /1,                               [DEFAUT]
                           /2,
◊TITRE=tx                                                       [kN]
);
```

3 Operands

3.1 Operands MODELE / CHAM_MATER / CARA_ELEM

♦MODELE = Mo

Nom of the model whose elements are the subject of mechanical computation.

♦CHAM_MATER = chmat

Nom of the affected material field on the model Mo. Attention, all the main meshes of the model must be associated with a material (if not fatal error with not very explicit message),

◊CARA_ELEM = carac

Nom of the characteristics (carac) of the shell elements, beam, pipe, bar, cable, and discrete elements affected on the model Mo. Obviously, this key word is optional: if models it does not contain such elements, it is not useful; on the other hand, if models it contains such elements, it is compulsory.

3.2 Key word EXCIT

♦ EXCIT=_F ()

This key word factor makes it possible to describe with each occurrence a load (stresses and boundary conditions), and possibly a multiplying coefficient and/or a kind of load.

3.2.1 Operands CHARGES

♦ LOAD = ch_i

ch_i is the mechanical loading (possibly comprising the evolution of a field of temperature) specified with i the ème occurrence of EXCIT.

Note:

1) *In a thermomechanical computation, if the initial temperature is different from the reference temperature (given in operator AFFE_MATERIAU), the strain field associated with initial time can be incompatible and thus lead to a stress state and intern variables associated non-zero. If one uses an incremental behavior model (key word factor COMP_INCR) and if one explicitly does not define a stress state and intern variables initial (associate at a field of initial temperature different from the reference temperature), the stress field and intern variables calculated with the first increment will take account only of the only variation in temperature between initial time and the first time, and not of the possible stresses of compatibility associated with the initial temperature. To take this initial state hopes some, it should be given explicitly, for example thanks to key words SIGM, DEPL, VARI in ETAT_INIT .* ***To avoid such situations which can lead to computational errors, it is to better begin a computation by considering that it is necessary to start from a virgin state.***
2) If one carries out a computation into axisymmetric and that one imposes nodal forces, these forces must be divided by 2π (one works on a sector of a radian) compared to the real loadings. In the same way, if one wishes compute the resultant of the forces, the result is to be multiplied by 2π having the total resultant on complete structure. In the same way in plane stresses or plane strain, one works on one thickness unit: the forces (on the thickness) applied must be divided by the thickness, the real forces are obtained by multiplying by the thickness the forces “of computation”.

3.2.2 Operand FONC_MULT

◊ FONC_MULT = f_i

f_i is the multiplying function of the time of the loading specified with the ième occurrence of EXCIT.
The loading and the boundary conditions pouroccurrences n of the key word factor EXCIT are:

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

For the conditions of Dirichlet, of course, only the specified value is multiplied par. f_i
By default: $f_i = 1$.

3.2.3 Operand TYPE_CHARGE

◊ TYPE_CHARGE =/"FIXE_CSTE" , [DEFECT]
/"SUIV",
/"DIDI",
/"FIXE_PILO"

Par defect, tchi is worth "FIXE_CSTE": that corresponds to a loading applied to the initial geometry and not controlled. It can however be a function, and, in particular, to depend on time.

If tchi is worth "FIXE_PILO", the loading is always fixed (independent of the geometry) but will be controlled thanks to key word PILOTAGE [§3.1221]. The loads controllable must result from AFFE_CHAR_MECA or AFFE_CHAR_MECA_F (if it is not a function depending on time) and do not be affected key word FONC_MULT. One cannot control the loadings of gravity, the centrifugal force, the forces of Laplace, the loadings thermal or of initial or unelastic strains, and the conditions of connection.

If tchi is worth "SUIV", the loading is known as "follower", i.e. it depends on the value of the unknown factors: for example, the pressure, being a loading applying in the normal direction to a structure, depends on the geometry brought up to date of the aforementioned, and thus on displacements. A following loading is revalued with each iteration of the algorithm of resolution. A fixed loading is revalued only at each new time, and only if chi depends on time (defined in AFFE_CHAR_MECA_F and parameterized by time).

Currently the loadings which can be qualified "SUIV" are the loading of gravity for the element of CABLE_POULIE, the pressure for modelizations 3D, 3D_SI, D_PLAN, D_PLAN_SI, AXIS, AXIS_SI, C_PLAN, C_PLAN_SI and for all the modelizations 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) and the centrifugal force in large displacements (key word ROTATION in AFFE_CHAR_MECA).

If tchi is worth "DIDI then " the conditions of Dirichlet (imposed displacements, linear conditions) will apply to the displacement increment as from the time given under ETAT_INIT/NUME_DIDI (by defect the time of resumption of computation) and not on total displacement. For example for a displacement imposed (key word DDL_IMPO of AFFE_CHAR_MECA) the condition will be form $u - u_0 = d$ where u_0 is the displacement defined by NUME_DIDI and not $u = d$.

3.3 Key word CONTACT

♦ CONTACT = contact

This key word simple makes it possible to activate the resolution of contact-friction or the taking into account of a unilateral connection. contact is a concept resulting from operator DEFI_CONTACT [U4.44.11].

Caution:
This simple key word accepts one concept. One cannot thus mix in the same nonlinear computation the resolution of the contact and the taking into account of a unilateral connection. One cannot mix either various formulations (DISCRETE, CONTINUE and XFEM)

3.4 Word-key SOUS_STRUC

◊SOUS_STRUC

This key word factor allows to specify which are the loadings to be used for under - static structures which then form obligatorily part of the model. In its absence, the loadings on under structures are null.

These loadings are added to the loadings "finite elements" which can be applied to the remainder of the model. For more accuracy concerning the use of substructures (elastic linear) in a nonlinear structure, one will refer to documentation [U2.07.02] and the ssnv193a benchmark.

3.4.1 Operand CAS_CHARGE

♦CAS_CHARGE = nocas

nocas is the name of the loading case to be used. See operator MACR_ELEM_STAT [U4.62.01].

3.4.2 Operands TOUT / SUPER_MAILLE

♦ /TOUT = "OUI"

This key word makes it possible to affect the loading nocas to all under structures of the model.

/SUPER_MAILLE = l_mail

This key word factor makes it possible to assign the loading nocas only to some under - structures.

3.4.3 Operand FONC_MULT

◊FONC_MULT = f_i

f_i is the multiplying function of the time of the loading specified with the ième occurrence of SOUS_STRUCT.
The behavior of this key word is the same one as for its occurrence in EXCIT.

3.5 Key words COMP_INCR and COMP_ELAS

the syntax of these key words common to several commands is described in the document [U4.51.11].

3.6 Key word ETAT_INIT

◊ **ETAT_INIT**

This keyword makes it possible to define an initial state of reference. By defaults, all the fields are identically null. The initial state can be defined either by specifying each field of the initial state, or in extraction since a concept of the preexistent evol_noli type. The data of an initial state does not have a meaning (and is not thus taken into account) only for the part of the field treated in incremental behavior (COMP_INCR); if the behavior is elastic (COMP_ELAS) that does not have any angle of attack.

If one wants to take into account an initial state in elasticity, it is key word ELAS located under COMP_INCR that it is necessary to use.
Note:

• If the user specified that the result concept is reentrant (by the word reserved reuse), key word ETAT_INIT is compulsory.
•If one uses the continuous method of the contact, the resumption of computation can give place to difficulties of convergence because of “the lapse of memory” of the state of preceding contact.

If an initial state is used from which the MODELE is different from the MODELE of computation indicated in the operator, Code_Aster proceeds automatically in the passing between the two models:

- If computation models it is included in the model given in the initial state, the data of the initial state are simply recopied mesh with mesh for all the fields;
- If models it given in the initial state is included in the model of computation, Code_Aster starts by copying the values of the initial state for the common meshes then complete with the zero value;

In the last case, it is necessary to take guard with the meaning which an initial null intern variable in computation can have.

3.6.1 Operands SIGM / VARI / DEPL / STRX

```
♦  / |  SIGM = sig
     |  VARI = vain
     |  DEPL = depl
     |  STRX = strx
```

sig is the stress field at the Gauss points, vain is the field of the intern variables at the Gauss points and depl is the field of displacements to the nodes taken in an initial state and strx is the field of forces and displacements corresponding to the structural elements. If one of these fields is not specified, it is taken null by default. They can for example result from command CREA_CHAMP, or be read in a file by command LIRE_RESU.

3.6.2 Operands EVOL_NOLI

```
/ EVOL_NOLI = evol
```

Nom of the concept of the evol_noli type from where will be extracted the initial state.

3.6.3 Operand NUME_ORDRE / INST / NUME_DIDI

```
◊ /NUME_ORDRE    = nuini
 /INST           = instini
```

Extraction of the initial mechanical state in evol starting from the number of filing NUME_ORDRE or of the time of filing INST to carry out the continuation of computation.
If NUME_ORDRE or INST are not filled, one takes the last existing number filed in evol.

```
◊  NUME_DIDI     = nudidi
```

Dans the case of loadings of differential the Dirichlet type (“DIDI”), one gives under NUME_DIDI the number of filing of the mechanical state (displacements) which is used as reference for the application of these boundary conditions. By defect one takes the definite mechanical state under NUME_ORDRE or INST.

3.6.4 Operand INST_ETAT_INIT

```
◊   INST_ETAT_INIT = istetaini
```

One can associate a value of time istetaini in this initial state.
By default:

1) When the initial state is defined by the data of fields (ETAT_INIT with DEPL/SIGM/VARI), it does not have there associated time.
2) When the state is given by a concept evol_noli (ETAT_INIT with EVOL_NOLI), it is time in preceding computation (istetaini = instini).

A - Simple Exemple (behavior by defect)

```
LIST1   = DEFI_LIST_REEL (DEBUT =0.,
                          INTERVALLE =_F (JUSQU'À = 4. , NOMBRE =4)),

U       = STAT_NON_LINE (INCREMENT =_F (LIST_INST =LISTE1)) ,

LIST2   = DEFI_LIST_REEL (DEBUT =4.,
                          INTERVALLE =_F (UNTIL = 10. , NOMBRE =6)),

U       = STAT_NON_LINE (reuse=U,
                         INCREMENT =_F (LIST_INST =LISTE2),
                         ETAT_INIT =_F (EVOL_NOLI =U)) ,
```

First STAT_NON_LINE: carry out computation for times 1 2 , 3 and 4s .
Second STAT_NON_LINE: carry out computation for times 5 6 7 8 , 9 and 10s , the initial state corresponding to time 4s .

B - Exemple to show the interest of INST_ETAT_INIT (two different lists of times)

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

U       = STAT_NON_LINE (INCREMENT =_F (LIST_INST =LISTE1)) ,

LIST2   = DEFI_LIST_REEL (DEBUT =20.,
                          INTERVALLE =_F (UNTIL = 30. , NOMBRE =10)),

U       = STAT_NON_LINE (reuse=U
                         INCREMENT =_F (LIST_INST =LISTE2),
                         ETAT_INIT =_F (EVOL_NOLI =U,
                         INST_ETAT_INIT = 20.)) ,
```

First STAT_NON_LINE: carry out the computation of times 1 with 10s .
Second STAT_NON_LINE: carry out the computation of times 21 with 30s , the initial state corresponding to the time $t=10\text{s}$ of the first STAT_NON_LINE (by default INST=10.). This initial state corresponds for this second STAT_NON_LINE to time $t=20\text{s}$. (INST_ETAT_INIT=20.).

C - Exemple to show the interest of INST_ETAT_INIT (practical when the cyclic one is made)

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

U1      = STAT_NON_LINE (INCREMENT =_F (   LIST_INST =LISTE1)) ,

U2      = STAT_NON_LINE (INCREMENT =_F (   LIST_INST =LISTE1),
                         ETAT_INIT =_F (   EVOL_NOLI =U1,
                                    INST_ETAT_INIT = 0.)) ,
```

First STAT_NON_LINE: carry out the computation of times 1 with 10s .
Second STAT_NON_LINE: carry out the computation of times 1 with 10s , the initial state corresponding to the time $t=10\text{s}$ of the first STAT_NON_LINE (by default INST=10.). This initial state corresponds for this second STAT_NON_LINE to time $t=0\text{s}$. (INST_ETAT_INIT= 0.).

3.6.5 Operand PRECISION / CRITERE

```
◊       PRECISION = prec
```

Confer [U4.71.00] for detailed syntax

This parameter is used to identify good sequence number (NUME_ORDRE) when the user informs time (INST). Indeed, times in STAT_NON_LINE are identified by a sequence number (an integer). If the user wants to use one time (a reality) and not a sequence number for INST, the operand accuracy makes it possible to select this sequence number.
Example:

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

If the user wants to select time corresponding to NUMÉRIQUE=4, it is enough for him to say INST=0,004. On the other hand , for the second example:

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

If the user wants to select time corresponding to NUMÉRIQUE=4, it is not enough for him to say INST=0,10000004, because the relative difference between times is worth $\frac{0,10000005-0,10000004}{0,10000004}=1\text{E-}7$ which is higher than the value of accuracy per defect ($1\text{E}-6$). One will not be able to thus distinguish NUMÉRIQUE=3,4 and 5 (the code stops then in fatal error). It is then enough to change parameter PRECISION to be able to select time (in the example, PRECISION=1E-8 will be appropriate).

3.7 Key word INCREMENT

♦ INCREMENT=_F ()

Définit intervals of time taken in the incremental method.

Times thus defined have physical meaning only for behavior models where time intervenes explicitly (viscoelastic or viscoplastic for example). In the other cases, they allow only indicer the increments of load and to parameterize the evolution of a possible field of temperature.

3.7.1 Operand LIST_INST

♦ LIST_INST= /litpsr8, [listr8]
/litps, [list_inst]

♦ If LIST_INST = litpsr8 [listr8]

Les times of computation are those defined in the concept litpsr8 by operator DEFI_LIST_REEL [U4.34.01].

♦ If LIST_INST = litps [list_inst]

Les times of computation are those defined in the concept litps by operator DEFI_LIST_INST [U4.34.03].

3.7.2 Operands NUME_INST_INIT / INST_INIT / NUME_INST_FIN / INST_FIN

/NUME_INST_INIT =nuini
/INST_INIT = instini

the initial time of the computation (which thus (Re) is not calculated) is indicated either by its value (INST_INIT), or by its sequence number in the list of times litps (NUME_INST_INIT). To be able to reach by value, it is necessary that the list is ordered.

In the absence of key words INST_INIT or NUME_INST_INIT, the defect is calculated in the following way:

1) If an initial state is specified (operand ETAT_INIT) and if it defines one time corresponding (by EVOL_NOLI or INST_ETAT_INIT) then time initial is that defined by this state initial,
2) If there is no initial state (operand ETAT_INIT absent) or that it does not define time corresponding (the fields are given in ETAT_INIT without specifying INST_ETAT_INIT), then one takes the first time of the list of times (NUMÉRIQUE_INST_INIT=0).
3) In the event of filing (see keyword ARCHIVAGE), initial time in continuation is the last pitch filed and not that defined in INST_INIT.

```
/NUME_INST_FIN   =nufin
/INST_FIN        =  instfin
```

final time (last calculated pitch) is indicated same manner that initial time (either NUME_INST_FIN, or INST_FIN), except that it is not possible to refer to the time of the initial state.

Caution:

- If the automatic recutting of the time step is activated, NUME_INST_FIN does not hold account and always works of it on the list of times initial. NUME_INST_INIT and NUME_INST_FIN are active only with initialization.

A - Simple Exemple (behavior by defect)

```
LISTE = DEFI_LIST_REEL (   DEBUT =0.,
                           INTERVALLE =_F (JUSQU'À = 10. , NOMBRE =10)),

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

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

First STAT_NON_LINE: carry out computation for times 1 2, 3 and 4s.
Second STAT_NON_LINE: carry out computation for times 5 6 7 8, 9 and 10s, the initial state corresponding to time 4s. (by default INST_INIT=INST_ETAT_INIT=INST=4.).

B - Exemple to show the interest of INST_INIT

```
LISTE = DEFI_LIST_REEL (   DEBUT =0.,
                           INTERVALLE = _F (JUSQU'À = 10. , NOMBRE =10)),

U = STAT_NON_LINE    ( INCREMENT = _F ( LIST_INST  = LISTE,
                                        INST_FIN   = 4.)) ,
U = STAT_NON_LINE    ( reuse = U,
                       INCREMENT =_F (  LIST_INST =LISTE,
                                        INST_INIT =8.),
                                        ETAT_INIT =_F (  EVOL_NOLI =U)) ,
```

First STAT_NON_LINE: carry out the computation of times 1 with 4s.
Second STAT_NON_LINE: carry out computation for times 9 and 10s (does not do anything for $t=5,6,7$ and 8s), the initial state corresponding to time $t=4\text{s}$ (by default INST=4.).

3.7.3 Operand PRECISION

◊ PRECISION = prec

Cf. [U4.71.00] for detailed syntax

This parameter is used to identify good sequence number (NUME_INST_FIN/NUME_INST_INIT) when the user informs time (INST_FIN/INST_INIT). Indeed, times in STAT_NON_LINE are identified by a sequence number (an integer). If the user wants to use one time (a reality) and not a sequence number for (NUMÉRIQUE_INST_*), the operand accuracy makes it possible to select this sequence number.

Example:

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

If the user wants to select time corresponding to NUMÉRIQUE=4, it is enough for him to say INST=0,10000004. On the other hand , for the second example:

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

If the user wants to select time corresponding to NUMÉRIQUE=4, it is not enough for him to say INST=0,10000004, because the relative difference between times is worth $\frac{0,10000005-0,10000004}{0,10000004}=1\text{E-}7$ which is higher than the value of accuracy per defect ($1\text{E-}6$). One will not be able to thus distinguish NUMÉRIQUE=3,4 and 5 (the code stops then in fatal error). It is then enough to change parameter PRECISION to be able to select time (in the example, PRECISION=1E-8 will be appropriate).

3.8 Operand CRIT_QUALITE

◊ **ERRE_TEMPS_ THM= /"NON" [DEFECT]**
/"OUI"

Cet operand makes it possible to activate the computation of the indicators of error. ERRE_TEMPS_THM is the temporal indicator of error for non stationary modelizations HM. See [R4.10.05].

3.9 Operand METHODE

◊ **METHODE = /"NEWTON"**
/"IMPLEX"
/"NEWTON_KRYLOV"

Permet to choose the method of problem resolution incremental nonlinear.

/"NEWTON"

One uses the algorithm of Newton-Raphson to solve the problem (see [R5.03.01]).

/"IMPLEX"

One uses algorithm IMPLEX to solve the problem (see [R5.03.81]).

/"NEWTON_KRYLOV"

One uses an inaccurate version of the algorithm of Newton-Raphson; the accuracy of the resolutions of systems linear by an iterative method is adapted during each pitch of loading (see [R5.03.01]).

3.10 Key word NEWTON

Précise characteristics of the method of resolution of the nonlinear incremental problem (method of Newton-Raphson)

3.10.1 Opérande PREDICTION

◊ **PREDICTION =/"TANGENTE"**
/"ELASTIQUE"
/"EXTRAPOLE"
/"the purpose of DEPL_CALCULE"

the phase of prediction (Cf. [R5.03.01]) is calculating an estimate of the field of displacements in order to make it possible the method of Newton more quickly to converge.
When the key word is absent, it is the tangent matrix of velocity (option RIGI_MECA_TANG) which is used if one chose for the method of Newton a MATRICE='TANGENTE', and it is the elastic matrix (option RIGI_MECA) which is used if MATRICE= were chosen'ELASTIQUE'.

/"TANGENTE"

One uses the tangent matrix of the problem of velocity (option RIGI_MECA_TANG).

/"ELASTIQUE"

One uses the elastic matrix (option RIGI_MECA).

/"EXTRAPOLE"

One computes the estimate of the displacement increment starting from the total increment obtained like solution with the preceding time step (balanced by the report of the time steps). One projects this estimate on all the kinematically admissible fields (i.e satisfying the boundary conditions with Dirichlet) according to the norm given by the elastic matrix, which must thus be computed. This functionality is interesting in the case of the use of diagrams of explicit integration local of Runge-Kutta type which do not provide a tangent matrix: in this case the method of Newton uses an elastic matrix, but the iteration count necessary can be high. The use of extrapolation can improve the performances.

/"DEPL_CALCULE"

Permet to propose like displacement for the prediction with each time step, the displacement given by a mechanical history specified under key word EVOL_NOLI (§1414). Displacement is project on kinematically admissible all the fields, as for method EXTRAPOLE.

Note:

- Methods "EXTRAPOLE" and "DEPL_CALCULE" carry out a projection of the solution on all the kinematically admissible fields. One helps oneself for that of the boundary conditions of Dirichlet given in keyword EXCIT. In this case, it is not possible to use loadings of "kinematical" Dirichlet of the type (operand AFFE_CHAR_CINE) but only of the loadings of Dirichlet by dualisation (operand AFFE_CHAR_MECA). An alarm warns the user if *Code_Aster* would not have found loadings of Dirichlet dualized. The risk in this case being that the field of displacement is not kinematically admissible.
- Because of this projection on the limiting conditions, these two options are incompatible with the functionalities of the PILOTAGE ;

Utility:

1) let us suppose that one carries out the first computation with a coarse mesh. One wishes to carry out same computation but on a finer mesh. One can suppose that the solution in displacement for this second computation is not distant from that of the first computation and thus that a good prediction of displacement for this second computation is the projection of displacements of the first computation on the nodes of the new mesh (the projection of displacements on the new mesh must be realized beforehand with operator PROJ_CHAMP [U4.72.05]). This key word makes it possible to carry out this mode of prediction.
2) that makes it possible to reduce the place memory and to preserve these results for a later continuation. For a large computation, one can store only displacements at all times with formats IDEAS or MED in IMPR_RESU. If one wants to recompute the stresses and intern variables, one makes a LIRE_RESU with the adequate format then one uses DEPL_CALCULE with ITER_GLOB_MAXI=0 (only one iteration is carried out) and ARRET='NON' (there is not convergence, one does not check the equilibrium). It is however necessary for reasons of syntax to give a loading (avoid

the Dirichlet loadings which impose a linear resolution) as well as a convergence criterion, even if this information is not taken into account.

3.10.2 Operand STAMPS

```
◊   MATRIX   =   / "TANGENTE"
◊   REAC_INCR = /1                    [DEFAUT]
                /mf
◊   REAC_ITER = /0                    [DEFAUT]
                /it
```

The matrix used for the total iterations of the method is the tangent matrix [R5.03.01]. The tangent matrix of prediction is revalued all MF increments of time (MF positive or null) and the coherent tangent matrix (option FULL_MECA) is revalued all the it iterations of Newton for an increment of time given (precisely with the iterations of number it, 2it, 3it...). Thus with the first iteration of Newton, one reassembles the tangent matrix only if it is worth 1: if not one keeps the matrix used in the phase of prediction. By convention if it is worth 0 the matrix is not revalued during all the time step.

```
◊  PAS_MINI_ELAS  = /0.                  [DEFAUT]
                    /pasmini          [R]
◊REAC_ITER_ELAS   = /0                 [DEFAUT]
                    /it               [I]
```

Ces options make it possible to pass from the tangent matrix to the matrix of discharge (i.e by considering that nonthe linearities do not evolve) when the time step is lower than pasmini. This matrix of discharge is the elastic matrix for the models of behavior of the plastic type; for the models of damage it is identified with the secant matrix.
As convergence with the elastic matrix is slower than that with the tangent matrix, key word ITER_GLOB_ELAS under the key word factor CONVERGENCE makes it possible to define an iteration count maximum specific to the use of the matrix elastic and different from that associated with the use of the tangent matrix.
One can define a frequency of reactualization of the matrix of discharge with key word REAC_ITER_ELAS (analogue of REAC_ITER). If the matrix of discharge does not depend on the strain state (what is the case for the plastic materials but not for the models of damage), take REAC_ITER_ELAS = 0 (since it will be the same one during iterations).

Utility:

This option can be useful when the automatic recutting of the time step is not enough to make converge a computation. For example, in the case of lenitive models, the tangent matrix can become singular and it is thus to better use the elastic matrix to converge.

```
◊   MATRIX   =   /"ELASTIQUE"
```

the matrix used corresponds to the elastic design: it is evaluated only once at initial time, at the beginning of algorithm. This "elastic" matrix is calculated by using the Young modulus given under key word ELAS of operator DEFI_MATERIAU, and not the slope at the origin of the traction diagram given under the key word TENSION (and which is useful, it, in the form of behavior models VMIS_ISOT_TRAC, VMIS_ECMI_TRAC, VISC_ISOT_TRAC [U4.51.11]).

3.10.3 Operand EVOL_NOLI

```
◊   EVOL_NOLI = evol_noli
```

Nom of the concept of the evol_noli type which will be useful in the prediction by DEPL_CALCULE.

3.11 Key word RECH_LINEAIRE

◊RECHERCHE_LINEAIRE=_F ()

linear search can make it possible to improve convergence of the method of Newton (Cf. [R5.03.01] for more details).

Caution:

It is disadvised using linear search with strains GROT_GDEP for modelizations COQUE_3D and in the presence of contact.

3.11.1 Operand METHODE

```
◊METHODE    = /"ROPE"                          [DEFECT]
              /"MIXTE"
              /"PILOTAGE"
```

Permet to choose the method of search linear, i.e. the algorithm of search of the zero of the functional calculus (see Doc. [R5.03.01]). The method TWISTS (by defect) is the simplest method, it is a unidimensional secant method.

Method MIXTE is worked out and uses a secant method with variable limits. It is more effective when the functional calculus is not strictly concave (problems with damage or THM for example).

Method PILOTAGE is reserved for control of the type DEFORMATION, PRED_ELAS and LONG_ARC (see §2121). It is the only method usable with this kind of control. For the control of the type DDL_IMPO, one can use ROPE or MIXTE.

3.11.2 Operand RESI_LINE_RELA / ITER_LINE_MAXI

```
◊RESI_LINE_RELA   = /1.E-1            [DEFAUT]
                    /reslin
◊   ITER_LINE_MAXI =   /3                [DEFAUT]
                    /itelin
```

They are the parameters of linear search. One gives the maximum iteration count itelin to be carried out and the accuracy reslin to reach to carry out the convergence of linear search. It is advised not to use linear search with contact.

For the method TWISTS, It is not necessary to specify an accuracy nor an iteration count very high, the practice showing that two or three iterations of linear search are sufficient. One can thus be satisfied to ask for three iterations with the accuracy by default. The user cannot put more than 999 iterations of linear search for the method TWISTS.
On the other hand, for method MIXTE, on problems with damage, several tens of iterations are often effective.

3.11.3 Operands RHO_MIN / RHO_MAX / RHO_EXCL

```
◊RHO_MIN=          /   1.E-2            [DEFAUT]
                   /   rmin             [R]
◊RHO_MAX=          /   1.E+1            [DEFAUT]
                   /   rmax             [R]
◊RHO_EXCL=             /   9.E-3            [DEFAUT]
                   /   rexc             [R]
```

Ces key words fixes the interval I in which one calculates coefficient RHO of linear search, in the form: $I=[rmin, rmax]-[-rexc, rexc]$ [R5.03.01].

3.12 Key word PILOTAGE

```
◊   PILOTAGE =_F ()
```

Lorsque the intensity η of part of the loading is not known a priori (loading known as of reference defined in AFFE_CHAR_MECA or AFFE_CHAR_MECA_F with load of the type FIXE_PILO), key word PILOTAGE makes it possible to control this loading via a node (or node groups) on which one can impose various modes of control (key word TYPE).

Caution:

• With FIXE_PILO , one cannot use for the loading of reference key word FONC_MULT .
• When the loading of reference is defined by AFFE_CHAR_MECA_F , this loading can be a function of the variables of space but not of time. In the same way, changes resulting from command variables (as the temperature) which depend on time are not usable with control.
• Key word PILOTAGE is interdict with the contact (except in the case of contact XFEM).
• It is not possible to make PILOTAGE with PREDICTION='DEPL_CALCULE' or PREDICTION='EXTRAPOLE' (see § 19)

3.12.1 Opérande TYPE

```
◊   TYPE=     /"DDL_IMPO"
              /"LONG_ARC "
              /"ANA_LIM"
              /"DEFORMATION
              /"PRED_ELAS"
              /"SAUT_IMPO"
              /"SAUT_LONG_ARC"
```

It is the type of control carried out. Seven modes of control are available (Confer [R5.03.80] for more details):

/"DDL_IMPO"

Permet to impose a given value of displacement increment (only one possible i component) in a single node No (or of a nodes group comprising one node). With each increment of time, one seeks the amplitude η of the loading of reference which will make it possible to satisfy the following incremental relation:

$$\text{cmult}.\Delta u_i(no)=\Delta t$$

/"SAUT_IMPO"

Reprend the principle of DDL_IMPO but to control the increment of the jump of displacement enters the lips of a crack X-FEM. Only one direction i is possible, but it can be defined in a local base (normal or tangent with crack). One controls the average of this increment of jump on a set of points of intersection P_a of the application interface with the edges a of the mesh. This group describes all crack if GROUP_NO is not indicated (behavior by defect), and only one part if it is it. Attention, this kind of control can be used only in modelization X-FEM.

$$\text{cmult}.\frac{1}{N}\sum_{a=1}^{N}[\![\Delta u_i]\!](P_a)=\Delta t$$

/"LONG_ARC"

Permet to control the intensity η of the loading of reference by the length (curvilinear abscisse) of the response in displacement of a nodes group (to be used for example when one wants to control the buckling of a test-tube). The following relation is checked:

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

where *N* are the nodes of control and *C* the components of the displacement of the nodes considered. Even if the group of node of control is tiny room to only one node, it is necessary nevertheless to use GROUP_NO.

/"SAUT_LONG_ARC"

Reprend the principle of LONG_ARC but to control the norm of the increment of the jump of displacement enters the lips of a crack X-FEM. One controls this norm on average on a set of points of

intersection P_a of the application interface with the edges a of the mesh. This group describes all crack if GROUP_NO is not indicated (behavior by defect), and only one part if it is it.

$$\text{cmult}.|||[\![\bar{\Delta u}]\!]|||=\Delta t \text{ with } |||[\![\bar{\Delta u}]\!]|||=\sqrt{\frac{1}{N}\sum_{c}\sum_{a=1}^{N}\left([\![\Delta u_c]\!](P_a)\right)^2}$$

where C are the components of displacement. Even if the group of node of control is tiny room to only one node, it is necessary nevertheless to use GROUP_NO.

/"ANA_LIM"

This mode of control is specific with the computation of Yield-point load (model NORTON_HOFF) by kinematical approach (cf [R7.07.01] for more detail). If $\boldsymbol{F}$ indicates the controlled assembled loading, TYPE_CHARGE='FIXE_PILO", then the function of control is written simply:

$$P(\boldsymbol{u})=\boldsymbol{F}.\boldsymbol{u}=1$$

Except for the computation of Yield-point load, this functionality is not of interest *a priori*. For this mode of control, no other key word is to be specified.

The use of lenitive constitutive laws can lead to brutal snap-backs which make delicate the course of computation. The two following modes of control cure it (Cf. [R5.03.80] for more detail).

/"DEFORMATION"

DEFORMATION guarantees that at least a point of Gauss of structure sees his strain evolving in a monotonous way. The relation is checked:

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

This mode of control is valid for all the constitutive laws including in large deformation SIMO_MIEHE.

/"PRED_ELAS"

PRED_ELAS ensures that at least a point of Gauss of structure left the threshold of elasticity linearized $f_{\text{pred-elas}}$ a quantity $\frac{\Delta t}{\text{cmult}}$. The relation is checked:

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

This mode of control is valid only for models ENDO_FRAGILE (with the local version and version nonlocal GRAD_EPSI), ENDO_SCALAIRE (with the nonlocal version), ENDO_ISOT_BETON and ENDO_ORTH_BETON (with the local version and the nonlocal version), BARENBLATT, BETON_DOUBLE_DP, CZM_EXP (with the elements with internal discontinuity *_ELDI) , CZM_OUV_MIX and CZM_TAC_MIX (elements of interface *_INTERFACE), CZM_EXP_REG (elements of joint *_JOINT or modelization X-FEM) and CZM_LIN_REG (elements of joint).
The fixing of the parameter cmult is difficult to define first blow because the concept of output of criterion $\frac{\Delta t}{\text{cmult}}$ is not intuitive and varies according to the constitutive laws. For models ENDO_FRAGILE, ENDO_SCALAIRE and ENDO_ISOT_BETON, a version different from the definition from $\frac{\Delta t}{\text{cmult}}$ is used, where this parameter is related to the increment of damage (see [R7.01.04]).

Use - Attention:

When one wants to use these the last two modes of control, it is essential to make a first STAT_NON_LINE without key word PILOTAGE to start the problem and to obtain an initial state ε^- different from zero (if not division by zero for control by increment of strain). One carries out after a recovery starting from this non-zero initial state and one uses control.
Moreover, the resolution of the two preceding equations makes it possible to obtain the unknown intensity of the loading. In certain cases, the solution of these equations can lead to several

solutions for the intensity. One then chooses always the solution which is closest to ε^-. This is why, when one wants to impose an alternated loading, one is obliged with each change of sign of the loading to carry out a first STAT_NON_LINE without key word PILOTAGE in order to obtain an initial compactness ε^- of tension or. One carries out then a second STAT_NON_LINE in continuation starting from the preceding initial state with key word PILOTAGE.

Note:

DEFORMATION and PRED_ELAS are not available for the structural elements.

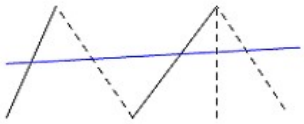
3.12.2 Operands NODE/GROUP_NO

```
/ NODE        =  No
/ GROUP_NO    =grno
```

to be used only with "DDL_IMPO", "LONG_ARC", "SAUT_IMPO" or "SAUT_LONG_ARC". For "DDL_IMPO", if operand GROUP_NO is used, the nodes group in question should contain one node. In the other cases, one uses only GROUP_NO (which can possibly contain one node). For "SAUT_IMPO" and "SAUT_LONG_ARC", the operand is optional.
For "DDL_IMPO" and "LONG_ARC", one gives the name of the node or the name of the nodes group on which one will impose control.

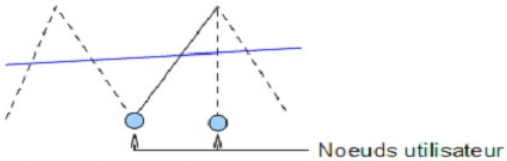
For "SAUT_IMPO" and "SAUT_LONG_ARC", the definition is more subtle since in modelization X-FEM one does not control the values on nodes but on points of intersection between the edges of the mesh and crack. In the continuation, one simply indicates by "edges" the intersected edges. The algorithm starts by building a set of independent edges which covers all crack (see fig.3.12.2-13.12.2-1). By default, it controls on all these edges. Key word GROUP_NO makes it possible to the user to restrict this group, each node well informed corresponding then at the end of an edge which one wishes to control. Let us announce the following rules then:

- if two nodes are the respective ends of two nonindependent edges, only one will be retained (fig. 3.12.2-2),
- if a node is end of several edges, one arbitrarily retains the first met by the algorithm,
- if two nodes are ends of the same crack (fig.3.12.2-33.12.2-3) an error is returned. Generally it is advised that all the entered nodes are same side of crack;
- if a node does not correspond to any edge (fig. 3.12.2-4), an error is returned.



Sélection automatique

Figure 3.12.2-1: GROUP_NO nonwell informed



Sélection utilisateur

Figure 3.12.2-2: Nodes ends of nonindependent edges

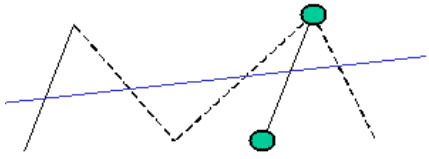


Figure 3.12.2-3: error for nodes connected to the same edge

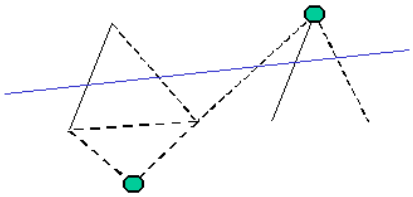


Figure 3.12.2-4: error for node off-line to an intersected edge

Tableau 3.1.

3.12.3 Operands TOUT/NET/GROUP_MA

```
 / TOUT=            "OUI"              [DEFECT]
/ GROUP_MA=lgrma
/ MAILLE=lma
```

One gives the meshes or mesh groups being used to control computation. To use only with DEFORMATION or PRED_ELAS. Interesting to reduce the resolution of the equations of these three modes of controls.

3.12.4 Operand NOM_CMP

◊ NOM_CMP = nomcmp

It is the name of the component (corresponding to the degree of freedom i) used for control ("DX" for example). To use only with "DDL_IMPO" or "LONG_ARC".

3.12.5 Operand DIRE_PILO

◊ DIRE_PILO = direpilo

It is the name of the direction i according to which one controls the jump of displacement. The possible values are: "DX", "DY", "DZ", "DNOR" for the norm with crack, "DTAN" for the first tangent (cross product of the norm with X), "DTAN2" for the second tangent. To use only with one modelization X-FEM. Use for types "SAUT_IMPO", "SAUT_LONG_ARC" or with "PRED_ELAS" if the selection on the choice of the controlled solution is "ANGL_INCR_DEPL" or "NORM_INCR_DEPL".

3.12.6 Operand FISSURES

◊ CRACK = fiss

Nom of the sd_fiss_xfem. To use only with one modelization X-FEM. Use for types "SAUT_IMPO", "SAUT_LONG_ARC" or with "PRED_ELAS" if the selection on the choice of the controlled solution is "ANGL_INCR_DEPL" or "NORM_INCR_DEPL".

3.12.7 Operand COEF_MULT

◊ COEF_MULT = cmult

It is the value (noted c_{mult} in the formula of definition) by which one multiplies the degree of freedom used for control. By default, this value is worth 1. Not to use with ANA_LIM.

Example with DDL_IMPO:

Let us suppose that one wants to know the Yield-point load of a structure.

The loading imposed on structure is the pressure of unknown intensity ($P=\eta\times$ valeur de référence P_x) on the group of mesh A . To find the Yield-point load P_{limite} , one will control the displacement of the node $NO1$. It is wanted that the following final displacement x of this node either equal to $2.$ (or according to the list of times of the pitches of 0.2 , or a coefficient $cmult=1/0.2=5.$)

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

LISTE=     DEFI_LIST_REEL (DEBUT=0          .,
                           INTERVALLE=_F (UNTIL = 10,  NOMBRE =10),

RESU=      STAT_NON_LINE (EXCIT=_F           ( LOAD      = PRESSURE,
                                               TYPE_CHARGE ='FIXE_PILO'),
                          PILOTAGE=_F        ( TYPE       = "DDL_IMPO",
                                               NODE       = NO1,
                                               NOM_CMP     = "DX",
                                               COEF_MULT  = 5.))
```

In the file .resu, the value of η will be at every moment displayed computation. To know the Yield-point load, it is enough to make $P_{limite}=\eta\times P_x$. (Here P_x 1 thus one is worth has the Yield-point load directly). If one imposes on structure a pressure P close to the Yield-point load without using control, computation will not converge if one is close to the Yield-point load.

Attention with the meaning of COEF_MULT for the control of the type PRED_ELAS.

3.12.8 Operand ETA_PILO_R_MAX / ETA_PILO_R_MIN

```
◊  ETA_PILO_R_MAX=etarmax     ,                [R]
◊ETA_PILO_R_MIN=etarmin       ,                [R]
```

Ces two key words make it possible to define the search interval of the values of control. With each iteration of Newton all the values of control apart from $[\text{etarmin},\text{etarmax}]$ are ignored. This can take along to "failure of control" if this interval is too restrictive.

If one does not specify values, it is $-\infty$ for etarmin and $+\infty$ etarmax. A possible use of this interval is the following. One wishes, for example, to control a pressure imposed on structure and one expects to keep this positive pressure. By fixing etarmin at 0, that makes it possible to impose the positive values of control.

3.12.9 Operand ETA_PILO_MAX/ETA_PILO_MIN

```
◊   ETA_PILO_MAX=etamax          ,              [R]
◊   ETA_PILO_MIN=etamin           ,             [R]
```

Ces two key words make it possible to specify the interval of values of desired control. Is used it to stop computation properly when ETA_PILOTAGE reaches the limits of this interval. This interval must be more restrictive than the search interval defined previously, because this last is applied in all the cases. The principle of operation is the following: with convergence of the iterations of Newton, if one reached one of the limits, computation stops. A possible use of this interval is the following one. In the case of presence of snap-back while fixing $etamin$ at a low value, that makes it possible to stop computation before a tear/complete damage of the sample and thus avoid the divergence with the last time step. The other possible use is that of $etamax$ as a maximum Yield-point load.

Caution:

With model ENDO_ISOT_BETON, these two key words are compulsory, because they are used to fix the limits of control at the elementary level.

3.12.10 Operand PROJ_BORNES

```
◊   PROJ_BORNES=    /"OUI"                     [DEFECT]
                    /"NON"
```

En cas de going beyond the interval (etamin, etamax), the user can indicate if he wants to project the value of control on (etamin, etamax).
With PROJ_BORNE='OUI', projection will be carried out (if eta>etamax then eta=etamax ; if eta<etamin then eta=etamin), which allows, in the event of convergence to stop computation precisely on etamin or etamax.
With PROJ_BORNE='NON', the values of eta are not modified, even if during the iterations of Newton the latter has a value higher than etamax or lower than etamin. On the other hand computation is stopped, so with convergence eta exceed the limits.

A possible use of the interval (etamin, etamax) with option PROJ_BORNE='OUI' is the following. One wishes, for example, to compare several computation for a lenitive model, which is to control in displacement. These parameters of control make it possible to stop computations with the same loading when the structure is sufficiently softened. This strategy makes easier the comparison, thanks to the check out of the last point of control.

With PROJ_BORNE='NON' one manages in certain cases to resolve computations, which differently do not converge with the too restrictive conditions imposed via (etarmin, etarmax). Either one controls a pressure imposed on structure and one expects to keep this positive pressure. By fixing etarmin at 0 computation stops in failure of control. On the other hand by imposing etarmin slightly negative, one authorizes de facto the transition by a state "not physics" during the iterations of Newton, which facilitates convergence. The state converged in this case can as well be physical (positive pressure) or not physical. It is the value of etarmin=0, which will control the behavior in the event of convergence except limit. This strategy makes it possible to preserve only the positive values of control, if one finds at least a value of control positive.

3.12.11 Operand SELECTION

```
◊  /SELECTION  =  /"NORM_INCR_DEPL",      [DEFECT]
                  /"ANGL_INCR_DEPL",
                  /"RESIDU"
                  /"MIXTE"
```

Cet operand makes it possible to select the method allowing for choice of the value of control if several solutions are provided by the resolution of control.
"NORM_INCR_DEPL" makes it possible to select the value of control by the smallest norm of the displacement increment on the time step considered.
"ANGL_INCR_DEPL" makes it possible to select the value of control by the smallest angle between the displacement obtained for the current time step and the displacement obtained for the preceding time step.
"RESIDU" makes it possible to select the value of control leading to the smallest residue.
"MIXTE" makes it possible to select the value of control by lean on several strategies. One starts initially with strategy "NORM_INCR_DEPL" above. If the results of the function objective (the norm of the displacement increment) are too close, one tilts for this iteration on strategy "RESIDU". There still, if the residues are too close, one returns to strategy "NORM_INCR_DEPL" and one examines whether the list of residues "RESI_GLOB_MAXI" of the time step running presents cycles. If it is the case, it is the least good solution of "NORM_INCR_DEPL" which is selected for this iteration. If not, one chooses simply the best of both, even if they are not sufficiently contrasted.

Note:

> *If one makes a resumption of computation (reuse) with keyword SELECTION='ANGL_INCR_DEPL', it is important to keep in mind that this criterion requires* the two *preceding time steps. It will thus be necessary well to take care to correctly archive the results of the preceding computation to the risk obtaining false results. An alarm informs the user.*

3.12.12 Operand EVOL_PARA

```
◊  EVOL_PARA=          /"SANS"                    [DEFECT]
                       /"DECROISSANT"
                       /" CROISSANT "
```

Cet operand makes it possible to impose it to it growth or the decrease of the parameter of control.

3.13 Key word SOLVER

```
◊  SOLVER =_F ()
```

the syntax of this key word common to several commands is described in the document [U4.50.01].

3.14 Key word CONVERGENCE

```
◊  CONVERGENCE =_F ()
```

If none of the two operands following is present, then all happens like if: RESI_GLOB_RELA = 1.E-6.

3.14.1 Operand RESI_GLOB_RELA/RESI_GLOB_MAXI

◊ |RESI_GLOB_RELA =resrel , [R]

the algorithm continues the total iterations as long as:

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

where $\boldsymbol{F}^n$ is the residue of the iteration n and $\boldsymbol{L}$ the vector of the imposed loading and the reactions of bearings (Cf. [R5.03.01] for more details).

When the loading and the reactions of bearing become null, i.e. when $\boldsymbol{L}$ is null (for example in the case of a total discharge), one tries to pass from convergence criterion relative RESI_GLOB_RELA to absolute convergence criterion RESI_GLOB_MAXI. This operation is transparent for the user (alarm message transmitted in the file .mess). When the vector $\boldsymbol{L}$ becomes again different from zero, one passes by again automatically with relative convergence criterion RESI_GLOB_RELA.
However, this mechanism of tilting cannot function with the first time step. Indeed, to find a value of RESI_GLOB_MAXI reasonable in an automatic way (since the user did not inform it), one needs to have had at least a pitch converged on a mode RESI_GLOB_RELA . Consequently, if the loading is null as of the first time, computation stops. The user must already then check that the null loading is normal from the point of view of the modelization which it carries out, and if such is the case, find another convergence criterion (RESI_GLOB_MAXI for example).

If this operand is absent, the test is carried out with the default value, except if RESI_GLOB_MAXI is present.

◊ |RESI_GLOB_MAXI =resmax , [R]

the algorithm continues the total iterations as long as:

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

where $\boldsymbol{F}^n$ is the residue of the iteration n (Cf. [R5.03.01] for more details). If this operand is absent, the test is not carried out.

If RESI_GLOB_RELA and RESI_GLOB_MAXI are present both, the two tests are carried out.

Note:

If the limiting conditions of Dirichlet are imposed by AFFE_CHAR_CINE (elimination) and not by AFFE_CHAR_MECA (dualisation), the degrees of freedom carrying these conditions are ignored during the rating of the residue of equilibrium. What does not cause false results but when the loading becomes null, i.e. when $\boldsymbol{L}$ is null (for example in the case of a total discharge), one passes from the convergence criterion relating to absolute convergence criterion RESI_GLOB_MAXI. This operation is transparent for the user (alarm message transmitted in the file .mess). When the vector $\boldsymbol{L}$ becomes again different from zero, one passes by again automatically with relative convergence criterion RESI_GLOB_RELA.

3.14.2 Operand RESI_COMP_RELA

◊ |RESI_COMP_RELA =rescmp , [R]

Cet operand results in considering the convergence of the algorithm of Newton while reasoning component by component. For that, one distinguishes in the vector residue the under-vectors corresponding to each component cmp (for example in THM, $cmp = \{DX, DY, DZ, PRE1, PRE2, TEMP\}$). One normalizes then these under-vectors by the corresponding internal force. Thus, the algorithm continues the total iterations as long as:

$$\max_{c=1,\ldots,nbcmp}\left(\frac{\max_{i=1,\ldots,nbddl}|\boldsymbol{F}_i^{c,n}|}{\max_{i=1,\ldots,nbddl}|\boldsymbol{L}_i^{\mathrm{int},c,n}|}\right)>\mathrm{rescmp}$$

where $\boldsymbol{F}^{c,n}$ is the part of the residue $\boldsymbol{F}^n$ corresponding to the component c and $\boldsymbol{L}^{\mathrm{int},c,n}$ the vector of the internal forces at time n corresponding to this same component c (Cf. [R5.03.01] for more details).

The internal forces at time n are computed at the beginning of time step starting from the result resulting from the preceding time step. For the first time step, one passes automatically to a relative criterion of type *RESI_GLOB_RELA*, to see with an absolute criterion for the cases where the loading is null. This choice has interest only for problems of the evolutionary type (THM) where resident of strong contrasts between the various unknown factors.

3.14.3 Operand RESI_REFE_RELA

```
|         RESI_REFE_RELA=resref  ,             [R]
   SIGM_REFE=sigref            ,              [R]
   FORC_REFE=              (forref, momref)       [l_R]
   VARI_REFE=varref            ,              [R]
   EPSI_REFE=epsref            ,              [R]
   FLUX_THER_REFE=fthref       ,              [R]
   FLUX_HYD1_REFE=fh1ref       ,              [R]
   FLUX_HYD2_REFE=fh2ref       ,              [R]
   DEPL_REFE          =  depref,              [R]
   LAGR_REFE          =  lagref,              [R]
```

Cet operand results in estimating the convergence of the algorithm of Newton in the following way. Starting from a reference, which can be:

- A stress `sigref` ;
- A strain `epsref` for incompressible elements, elements of grid and membrane;
- An intern variable `varref` if one uses nonlocal models with deformation gradient;
- A heat flux `fthref` in a case THM;
- Two hydrous flows `fh1ref` and `fh2ref` in a case HHM;
- A depref `displacement` if one uses elements of joint with a behavior of the type CZM;
- A force `forref` and a moment `momref` if structural elements are used (discrete, bars, beams or cables);
- A coefficient of Lagrange `lagref` for the mixed formulations in damage.

A reference of residue is computed F^{ref} (a of the same vector length than the vector residue). Convergence will be carried out if and only if:

$$\forall i\in[1,\ldots,nbddl]\ \ |\boldsymbol{F}_i^n|<\mathrm{resref}\,.\,\boldsymbol{F}_i^{ref}$$

3.14.4 Operand ITER_GLOB_MAXI

```
◊   ITER_GLOB_MAXI =   /10                  [DEFAUT]
                     /maglob
```

Maximum iteration count carried out to solve the total problem at every moment (10 by defaults). This test is always carried out except in the case of the recutting of the time step by method "EXTRAPOLE". The excessive increase in this parameter is generally the sign of a problem in the modelization or an inadequate temporal discretization.
In the case of the resolution of a problem of contact/friction by formulation `CONTINUE` in generalized Newton, it is often necessary to increase the iteration count of Newton.

3.14.5 Operand ITER_GLOB_ELAS

```
◊   ITER_GLOB_ELAS =   /25                  [DEFAUT]
```

/maxelas

Maximum iteration count carried out with the elastic matrix when one uses key word PAS_MINI_ELAS of the key word factor NEWTON (see §1818) to solve the total problem at every moment (25 per defect).

It is reminded the meeting that PAS_MINI_ELAS makes it possible to pass from the tangent matrix to the elastic matrix when the time step is or becomes (by recutting) lower than a certain value specified under PAS_MINI_ELAS.

Contrary to ITER_GLOB_MAXI, this parameter can easily take great values (several hundreds) because convergence on a nonlinear problem with the elastic matrix (very stiff) slow although is ensured from the theoretical point of view for all the models describing the generalized standard materials.

3.14.6 Operands TYPE/PLATEAU_ITER/PLATEAU_RELA

```
◊TYPE            =  /"PIC"              [DEFECT]
                    /"PLATEAU"
◊PLATEAU_ITER=         /3                   [DEFAUT]
                    /plaite             [I]
◊PLATEAU_RELA=         /1E-3                [DEFAUT]
                    /plarel             [R]
```

Cet operand makes it possible to control the type of operator of convergence to be applied for keywords RESI_*. By defect, mode "PIC" ensures that there is convergence since the value threshold given by RESI_* is reached. Mode "PLATEAU" is more severe and requires that the convergence criterion be stable around during plaite iterations in a tunnel of width plarel around the value of reference given by RESI_*. This mode is useful in the modelizations THM for which the criterion of Newton is sometimes not severe enough.

3.14.7 Operand ARRET

```
◊ARRET  = /"OUI"                                    [DEFAUT]
```

If one of the selected convergence criteria total is not checked after maglob iterations, then the program stops (the preceding results are saved).

```
◊ARRET  = /"NON"                                    [DEFAUT]
```

If maglob is insufficient to check the convergence criteria given by the user, one passes nevertheless to next time. This option is usable only in mode DEPL_CALCULE.
This option is to be used with precaution because it gives false results.

3.15 Mot-clé CRIT_STAB

```
◊   CRIT_STAB =_F ()
```

This key word makes it possible to start computation, at the end of each increment of time, of a criterion of stability. This criterion is useful to detect, during the loading, the point from which one loses stability.

- By buckling in the case of mechanical phenomenon reversible:

This criterion is then computed in the following way: at the end of a time step, in small disturbances, one solves $det\left(K^T-\lambda\cdot K^g\right)=0$ · K^T is the coherent tangent matrix at this time. K^g is the geometrical stiffness matrix, computed starting from the stress field at this time.
In practice, the loading is unstable if $|\lambda|<1$ (in fact $-1<\lambda<0$). One computes the eigenvalues by the method of Sorensen (*cf* MODE_ITER_SIMULT [U4.52.03]). This can be expensive enough for the problems of big size. For large displacements and the large deformation, one solves $det\left(K^T-\lambda\cdot I\right)=0$ because K^T contains then K^g. The criterion is then a criterion of instability:

when λ changes sign (thus passes by 0) the loading is unstable. One stores the eigen mode the corresponding to smallest critical load (in absolute value) in the result object, under name MODE_FLAMB . This eigen mode can be extracted and visualized (like a field of displacements or a conventional eigen mode). It is standardized to 1 on the largest component of displacement. The linear analysis of stability not making it possible to take account of the following aspect of certain forces, it is then necessary to use CRIT_STAB.

Documentation [U2.08.04] present the various approaches for the analyses of buckling in Code_Aster.

- By a study of sign on derivative second of energy by respecting the increase in the irreversible degrees of freedom in the case of dissipative mechanics:

MIXTE

Pour process this case particular, one forces to take as geometrical stiffness matrix the matrix identity $K^g = I_d$. One seeks then the minimum of following quadratic functional calculus:

$C(x) = \frac{x^t . K^T x}{x^t . x}$ where K^T is the coherent tangent matrix at studied time and x^t the vector transposed from the field of unknown nodal x , under stresses of positivity on the degrees of freedom of x irreversible nature. The sign of the minimum then makes it possible to conclude on the stability of the loading. If the aforementioned is negative, the solution is unstable. In the contrary case, the solution obtained numerically is stable. The mode obtained, which are the vector minimizing $C(x)$ (known as mode of instability if the minimum is negative), and the estimate of the associated criterion of stability are stored in the result object under name MODE_STAB (CHAR_STAB = $C(x)$).

3.15.1 Operand LIST_INST / INST / PAS_CALC

```
◊  /"LIST_INST'  = list_r8
   /"INST'       = l_r8
   /"PAS_CALC'   = npas
```

Les times for which one wants to make a computation of stability is given by a list of times (liste_r8 or l_r8) oupar a frequency PAS_CALC (all $npas$ of time).
In the absence of these key words the criterion is computed with all the time steps.

3.15.2 Operand PRECISION/CRITERE

```
◊   PRECISION=      /1.e-6                    [DEFAUT]
                   /prec
◊CRITERE=          /"RELATIF",              [DEFECT]
                    /"ABSOLU",
```

Permet to select times, *confer* [U4.71.00]

3.15.3 Opérande CHAR_CRIT

```
◊   CHAR_CRIT=      /(- 10.10),              [DEFAUT]
                   /intcc
```

Key word CHAR_CRIT makes it possible to save time by making only one test of Sturm in the provided frequency band. If at least a frequency is found, then one computes really the values of the critical loads in this interval.

3.15.4 Operand NB_FREQ

```
◊   NB_FREQ  =  / 3 ,                        [DEFAUT]
                / nbfreq
```

key word NB_FREQ (3 per defect) indicates the number of critical loads to calculating. In fact only the first is enough but there can be multiple modes.

3.15.5 Operand COEF_DIM_ESPACE

```
◊  COEF_DIM_ESPACE =  / 5,                    [DEFAUT]
                      / coeff
```

key word COEF_DIM_ESPACE (5 is its default value) makes it possible to the user to control the size of the subspace in the method of Sorensen (the size of under space is equal to the multiplication of this coefficient by the previously well informed value nbfreq). Interest being to be able to reduce this space if one uses in more operand DDL_STAB.

3.15.6 Operand RIGI_GEOM

```
◊  RIGI_GEOM = / "OUI",                       [DEFECT]
               / "NON"
```

key word RIGI_GEOM ("OUI" by defect) gives the choice to the user between or not carrying out a search for eigenvalues generalized with the geometrical matrix with the second member (case of the large deformation). To choose "NON" means that the geometrical stiffness matrix is replaced by the identity.

3.15.7 Operand MODI_RIGI

```
◊  MODI_RIGI = / "NON",                       [DEFECT]
               / "OUI"
```

key word MODI_RIGI ("NON" by defect) makes it possible to specify if the global stiffness matrix (and the geometrical stiffness matrix if it is used) must be modified on the level of the degrees of freedom which one lists with DDL_EXCLUS. That allows, for example, for mixed models, not to carry out the analysis of stability by excluding certain type of degree of freedom and by correcting the total stiffness matrixes so that the terms related to these degrees of freedom do not come to disturb the search for instability. One gives other details in the paragraph devoted to DDL_EXCLUS. If the list of excluded degrees of freedom is empty, then MODI_RIGI is not thus used for nothing.

3.15.8 Operand DDL_EXCLUS

```
◊  DDL_EXCLUS = ("DX", "DY",…)
```

Key word DDL_EXCLUS (empty list by defect) indicates all the degrees of freedom which one wishes to put at 0 in the second member of the search for generalized eigenvalues. It can be used only under the condition RIGI_GEOM = "NON" or if MODI_RIGI = "OUI".
In the case RIGI_GEOM = "NON" and MODI_RIGI = "NON", that makes it possible to impose additional conditions of compatibility on the eigen modes and thus to carry out a selective search. That is particularly adapted to the mixed formulations. In this case, the elimination of the Lagrange multipliers to the second member, makes it possible to exclude the parasitic modes with dominant Lagrangiennes and of negative eigenvalues.

In the case MODI_RIGI = "OUI" , that makes it possible to modify the stiffness matrix (and if need be the geometrical stiffness matrix) so as to carry out the analysis of stability by not holding account of the excluded degrees of freedom. For example, one must use this option for the models fluid-structure coupled (formulation (u, p, ϕ), *confer* documentation [R4.02.02], which is usable with DYNA_NON_LINE but not STAT_NON_LINE) to exclude the fluid degrees of freedom because the total assembled stiffness matrix is singular for these degrees of freedom. For more details, the user will be able usefully to refer to documentations [U2.06.11] and [U2.08.04].

3.15.9 Operand DDL_STAB

```
◊  D.O.F. _STAB = ("DAMG",…)
```

The key word D.O.F. _STAB indicates all the irreversible degrees of freedom in the study of stability which one wishes to carry out with CRIT_STAB . It can be used only under the conditions:

TYPE='STABILITY" and RIGI_GEOM='NON" . That makes it possible to carry out a study of sign on derivative second of energy, with the loading considered, by looking at only the disturbances likely to increase the degrees of freedom declared in DDL_STAB . This in order to observe the mechanical conditions of irreversibility.

3.15.10 Operand SIGNE

```
◊  SIGNE          =  /"POSITIF_NEGATIF",          [DEFECT]
                  =  /"POSITIF",
                  =  /"NEGATIF",
```

key word SIGNE makes it possible to specify which type of criterion of instability will be used. This criterion will make it possible to start a clean stop (bases saved) of nonlinear computation in the event of instability, if the user specifies it, under DEFI_LIST_INST (*confer* documentation [U4.34.03]) with following syntax:

```
ECHEC=_F (EVENEMENT='INSTABILITY", ACTION='ARRET',)
```

Without this statement under DEFI_LIST_INST , even in the event of detected instability nonlinear computation will try to continue: it is the by default mode.

For the analyses of stability without geometrical stiffness matrix, the criterion of instability it is that a critical load tends towards 0, or changes sign. In this case, key word SIGNE is not useful.

On the other hand, for the cases where the geometrical stiffness matrix is used, this key word SIGNE is useful. With the value by default: SIGNE = " POSITIF_NEGATIF" , the soluti one will be declared unstable whenever a critical load becomes ranging between -1 and 1. If the user chooses option "NEGATIF then " the field of instability will be limited by values -1 and 0. Conversely, option "POSITIF" will define values 0 and 1 like limits of the field of instability. The choice by defect is most conservative, but in certain cases where one can clear a priori part of the field of instability, then it is relevant to modify the criterion with key word SIGNE . It is reminded the meeting that the critical load computed by CRIT_STAB , if the geometrical stiffness matrix is taken into account, is the reverse of the multiplying coefficient of the imposed loading which makes the problem unstable. Thus if the computed value by CRIT_STAB is worth -1 that means that one is unstable for the imposed load. If value 1 is obtained, then instability will occur for an imposed load of the same value but of opposite sign. Thus for imposed loadings known and evolving in a monotonous way, it is easy to restrict the field of instability because it is known that the loading cannot change sign. On the other hand, for cyclic or unspecified loadings, it is surer not to restrict the field of instability.

3.15.11 Operand PREC_INSTAB

```
◊ PREC_INSTAB=      /1.E-6,                     [DEFAUT]
                    /prec_instab,               [R]
```

key word PREC_IN STAB makes it possible to define the relative tolerance with which one wishes to check the criterion of instability, which is parameterized by preceding key word SIGNE .

3.16 Key word ENERGIE

```
◊  ENERGIE =_F ()
```

This key word makes it possible to activate the computation of the assessment of energy, its display in the course of computation and its storage in the array of name PARA_CALC. The assessment of energy can be extracted from this array using command RECU_TABLE [U4.71.02].

3.17 Key word ARCHIVAGE

```
◊  ARCHIVAGE =_F ()
```

Permet to file or certain results at all or certain times of computation.

In the absence of this key word all the time steps are archived, including times of computations lately created by automatic recutting of the time step. Filing makes it possible to appreciably reduce the size of the bases by selecting saved times.

3.17.1 Operand LIST_INST / INST / PAS_ARCH

```
◊  /"LIST_INST' = list_r8
   /"INST'      = l_r8
   /"PAS_ARCH'  = npas
```

the designation of times to be stored is carried out either by a list of times (liste_r8 or l_r8) or then by a frequency of filing (all npas of time).
In the absence of these key words all the time steps are archived.

Two note:

1) *the last computation step is always stored to be able to carry out a recovery,*
2) *if one employs an access by list of times, then times of computations lately created by automatic recutting of the time step are not filed*
3) *the initial state is systematically filed under sequence number 0 since one is not in resumption of computation (not of reuse)*

3.17.2 Opérande PRECISION/CRITERE

```
◊  PRECISION=   /1.e-6          [DEFAUT]
                /prec
◊  CRITERE=     /"RELATIF",      [DEFECT]
                /"ABSOLU",
```

Cf. [U4.71.00]

3.17.3 Opérande CHAM_EXCLU

```
◊  CHAM_EXCLU
```

Permet to specify the fields which will not be filed, except with the last time step.
The name of the excluded fields depends on the operators.

3.18 Key word AFFICHAGE

```
◊  AFFICHAGE =_F ()
```

This keyword factor makes it possible to personalize the display of the table of convergence in STAT_NON_LINE and DYNA_NON_LINE.

If this keyword is not indicated, the table is built according to the various computation options (linear search, control, contact, etc) and with INFO_RESIDU='NON".

3.18.1 Operand UNITE

```
◊  UNITE =unit
```

the table of convergence will be duplicated in the file of unit *links*, with the format .csv (the separator being the comma).

3.18.2 Operand INFO_RESIDU

```
◊  INFO_RESIDU =  /"NON",                    [DEFECT]
                  /"OUI"
```

Cet operand makes it possible to add a column for each evaluated residue (RESI_GLOB_RELA, RESI_GLOB_MAXI, RESI_COMP_RELA and RESI_REFE_RELA). This column will indicate the node where the residue is maximum, which can help the user when there are difficulties of convergence. For example, to see whether the material were badly defined with an incorrect value on an element.

3.18.3 Operand INFO_TEMPS

```
◊  INFO_TEMPS =    /"NON",                    [DEFECT]
                   /"OUI"
```

Cet operand makes it possible to add a column which to give the time spent in the iteration of Newton.

3.19 Key word OBSERVATION

```
◊  OBSERVATION =_F ()
◊  TITLE        = title
```

This key word makes it possible post-to treat certain fields with the nodes or the elements on parts of model times of a list (known as of observation) generally more refined than the list of the times filed defined in the key word ARCHIVAGE [§3.1733] (where one stores all the fields on all models it). It is used primarily with economies as storage, but also to evaluate fields on reduced parts of the mesh, without needing post-to treat after computation. It is possible, for example, of compute the norm of the stresses, within the meaning of Von-Bet, and to store it in the array of observation.

This key word is répétable and allows the creation of an array of of the same observation name than the result concept of STAT_NON_LINE as one will be able to extract using command RECU_TABLE.
One can use only 99 occurrences of keyword OBSERVATION to the maximum.
It is possible to name an occurrence of the observation (column NOM_OBSERVATION) by using the keyword TITRATES. If it is not used, column NOM_OBSERVATION contains OBSERVATION_xx with xx variable from 1 to 99.

3.19.1 Operands LIST_INST / INST / PAS_OBSE

```
◊  /"LIST_INST' = list_r8
   /"INST'      = l_r8
   /"PAS_OBSE'  = npas
```

Ces operands makes it possible to define in the choices a list of times of observation. LIST_INST, INST and PAS_OBSE have the same meaning as the of the same operands name being used to define a list of filing. PAS_OBSE playing the same part as PAS_ARCH in ARCHIVAGE [§3.1733].

3.19.2 Operands PRECISION / CRITERE

```
◊   PRECISION = prec
◊   CRITERE   = /"ABSOLU"
                /"RELATIF"
```

Cf. [U4.71.00] for detailed syntax.
These parameters make it possible to manage the accuracy of the selection of times for the observation.

3.19.3 Operands NOM_CHAM / NOM_CMP

```
♦  NOM_CHAM  = nomcham
♦  NOM_CMP   = nomcmp
```

Ces operands make it possible to define the field post-to be treated (NOM_CHAM) like its components given by their name (NOM_CMP). One can define only 20 components to the maximum by occurrence of the keyword factor OBSERVATION.

3.19.4 Operands TOUT/NODE/GROUP_NOEUD/NET/GROUP_MA

```
◊  /TOUT          =  "NON"                          [DEFECT]
                      "OUI"
   /NOEUD         =  No                             [No]
   /GROUP_NO      =grno                             [grno]
   /MAILLE        =lma                                  [my]
   /GROUP_MA      =lgrma                             [grma]
```

Ces operands makes it possible to define the geometrical support of postprocessing:
- for fields with the nodes ("DEPL", "VITE", "ACCE", "DEPL_ABSOLU", "VITE_ABSOLU", "ACCE_ABSOLU", "VALE_CONT", "FORC_NODA"), one extracts the list from the nodes.
- for fields at the Gauss points ("SIEF_ELGA" , "VARI_ELGA"), one extracts the list from the meshes.

Attention not to use TOUT='OUI" on large meshes!

3.19.5 Observation of a field ELGA

```
◊  EVAL_CMP =   /" VALE ",                           [DEFECT]
                / "FORMULA"
◊  FORMULATES =   form                                  [ formule_aster ]
```

One starts by choosing the components or the formula between the components:
- If EVAL_CMP = " VALE " , one extracts simply the list from the components given by NOM_CMP .
- If EVAL_CMP = "FORMULA" , one evaluates the formula given by the word-key-simple FORMULA .

If one applies a formula to the components, one will thus have a value and thus an observation, if not, one will have as many observations as components in list NOM_CMP .

```
◊  EVAL_ELGA      =   /" VALE ",                      [DEFECT]
                      /" MIN ",
                       /" MAX ",
♦  /POINT=pi                                            [I]
◊  /SOUS_ POINT=spi                                  [I]
```

Une fois evaluated the components or the formula on the components, one can:
- Extract these values on the points and subpoints from integration with EVAL_ELGA = " VALE ". In this case, it is necessary to explicitly clarify the point and the subpoint of integration by POINT and SOUS_POINT . The subpoints of integration appear for structural elements (beams, plates, shells, pipes, etc).
- Ask to extract the maximum EVAL_ELGA = "MAX" or minimum EVAL_ELGA = "MIN" on all the points and subpoints of a mesh.

If one explicitly asks a point and for a subpoint, one achievements as of points requested, will have multiplied as many by the component count requested. On the other hand, if one asks the maximum or for the minimum, there is will have only one observation per component requested.

```
◊  EVAL_CHAM   =  /"VALE",                           [DEFECT]
                  /"MIN",
                  /"MAX",
                  /"MOY",
                  /"MINI_ABS",
                  /"MAXI_ABS",
```

"MINI_ABS " is the minimal value in absolute: MINI_ABS (- 1,3,4, - 12, - 0.1) = 0.1
"MAXI_ABS" is the maximum value in absolute: MAXI_ABS (- 1,3,4, - 12, - 0.1) = 12
Une fois evaluated the components (or formulates it on the components), have nsi that the point/subpoint of extraction, one can:
- Extract these values on all the meshes with EVAL_ CHAM= " VALE ".
- Ask to extract maximum EVAL_ CHAM= "MAX", minimum EVAL_ CHAM= "MIN" or average EVAL_ CHAM= "MOY" .

Example: Extract L E maximum from the trace of the tensor of the stresses on the GROUP_MA='TOTO "

```
 traces          = FORMULA (VALE='0.333* (SIXX+SIYY+SIZZ)",
                            NOM_PARA= ("SIXX", "SIYY", "SIZZ",));
OBSERVATION=_F    ( NOM_CHAM ='SIE F _ELGA',
                    GROUP_MA = "TOTO",
                    EVAL_CHAM = "MAX",
                    NOM_CMP = ("SIXX", "SIYY", "SIZZ",),
                    EVAL_CMP = "FORMULA",
                    FORMULATES = trace,
                    EVAL_ELGA = "MAX")
```

3.19.6 Observation of a field NOEU

```
◊   EVAL_CMP =   /" VALE ",                              [DEFECT]
                 / "FORMULA"
◊   FORMULATES  =   form                                    [ formule_aster ]
```

One starts by choosing the components or the formula between the components:

- If EVAL_CMP = " VALE " , one extracts simply the list from the components given by NOM_CMP .
- If EVAL_CMP = "FORMULA" , one evaluates the formula given by the word-key-simple FORMULA .

If one applies a formula to the components, one will thus have a value and thus an observation, if not, one will have as many observations as components in list NOM_CMP .

```
◊   EVAL_CHAM   =   /" VALE ",                           [DEFECT]
                    /" MIN ",
                    /" MAX ",
                    /" MOY ",
```

Une fois evaluated the components (or formulates it on the components), one can:

- Extract these values on all the meshes with EVAL_ CHAM= " VALE ".
- Ask to extract maximum EVAL_ CHAM= "MAX", minimum EVAL_ CHAM= "MIN" or average EVAL_ CHAM= "MOY" .

Example: Extract the maximum of component DX from displacement on GROUP_NO ='TOTO "

```
OBSERVATION=_F    ( NOM_CHAM =" DEPL ",
                    GROUP_NO = "TOTO",
                    EVAL_CHAM = "MAX",
                    NOM_CMP = (" DX ",),
                  )
```

3.19.7 Contenu of the array

the array will contain to the maximum 16 columns.

NOM_OBSERVATION	K80	Nom automatically given or by the key word TITRATES
TYPE_OBJET	K16	the array thus contains only actual values R
NOM_SD	K24	" "
NUME_REUSE	I	Indice of re-use of the array in the event of REUSE
NUME_OBSE	I	Sequence number of Urgent
observation	INST	R of observation
NOM_CHAM	K16	Nom of the field observed
EVAL_CHAM	K8	Type of evaluation of field
NOM_CMP	K8	Nom of component observed

EVAL_CMP	K8	Type of evaluation of the component
NODE	K8	Node where the observation (field with the nodes) is carried out
NETS	K8	Nets where the observation (field with the meshes) is carried out
EVAL_ELGA	K8	Type of evaluation of the fields at Gauss points
POINT	I	Point of integration where it is carried outobservation (fields with the meshes)
SOUS_POINT	I	Subpoint of integration where the observation (fields with the meshes) is carried out
VALE	R	Valeur

parameter NUME_REUSE is useful in the event of enrichment of the data structure result. Indeed, if the recovery crushes old sequence numbers in the data structure result (see keyword ETAT_INIT), it is not the case of the values in the array of observation, which is never modified retroactively. One can thus have two different values for same time in the array, the distinction will be done then on NUME_REUSE.

3.20 Key word SUIVI_DDL

◊ SUIVI_DDL =_F ()

This key word makes it possible post-to treat certain fields with the nodes or the elements on parts of model all the iterations of Newton and to display them in the table of convergence. The simultaneous number of SUIVI_DDL depends on the displayed columns and thus on the activated functionalities.

The key word factor SUIVI_DDL has same syntax as OBSERVATION for the extraction of the fields, except that one does not give information over times to be extracted, since one carries it out with each iteration of Newton (it does not have key words LIST_INST there/INST/PAS_OBSE/CRITERE/PRECISION).

◊ TITLE =ltitre , [list_k]

This key word expects a list of three warps to the maximum and makes it possible to name the column of the display board. The warps are truncated with 16 characters.

3.21 Contained data structure EVOL_NOLI

data structure EVOL_NOLI contains the list of the fields filed during computation (according to the various options of keyword ARCHIVAGE). By default, it contains, for each time, the list of the following fields:

- DEPL : field (with the nodes) of displacements;
- SIEF_ELGA : field (at the Gauss points) of the stresses;
- VARI_ELGA : field (at the Gauss points) of the intern variables;
- COMPOR : card of the behavior;

According to certain computation options, other fields will be present:

- VALE_CONT : field (with the nodes) of information on contact-friction (see [U4.44.11] for more details on the contents of this field);
- INDC_ELGA : field (at the Gauss points) of the statutes of contact for case XFEM with contact;
- COHE_ELGA : field (at the Gauss points) of the parameter of cohesion for case XFEM with RELATION='CZM" ;
- SECO_ELGA : field (at the Gauss points) of the statutes of friction for the case XFEM with contact and friction;

Besides these fields, the data structure also contains parameters. At every moment, one stores at least:

Name	Word-key origin	Description	Type
INST		Valeur of the time of computation	R
EXCIT	EXCIT	Informations on loadings	K24
MODELE	MODELE	elementary	Model
K8	CARAELEM	CARA_ELEM Caractéristiques	K8
CHAMPMAT	CHAM_MATER	Material field	K8
PARM_THETA	COMP_INCR/PARM_THETA	Paramètre of integration of constitutive law	R
ITER_GLOB		total Nombre of iterations of Newton	I
CHAR_MINI		minimum Chargement reached during time step	R
ETA_PILOTAGE		Paramètre of control	R

Quand one seeks modes of instability (with STAT_NON_LINE or DYNA_NON_LINE) or of the oscillatory modes (with DYNA_NON_LINE only), one stores the field of corresponding displacement and the value of the critical loading or the frequency.

Name	Word-key origin	Description	Type
CHAR_CRIT	CRIT_STAB with TYPE = "FLAMBEMENT"	critical Loading of the mode of buckling	R
MODE_FLAMB	CRIT_STAB with TYPE = "FLAMBEMENT"	Mode of buckling	Field of the type DEPL
CHAR_STAB	CRIT_STAB with TYPE = "STABILITY"	Valeur of instability	R
MODE_STAB	CRIT_STAB with TYPE = "STABILITY"	Mode of instability	Field of the type DEPL
FREQ	MODE_VIBR	Frequency of oscillatory mode (just available in DYNA_NON_LINE)	R
DEPL_VIBR	MODE_VIBR	Oscillatory mode (just available in DYNA_NON_LINE)	Field of the type DEPL

3.22 Opérande INFO

◊ INFO =inf

Permet to carry out in the file message various intermediate printings.
Other printings are made systematically during nonlinear computation, independently of the value assigned to key word INFO : they are the printings of the residues and the relative increments of displacement during iterations of Newton.
Attention, the files .mess can become very important with INFO = 2.

3.23 Operand TITRATES

◊ TITLE = tx

tx is the title of computation. It will be printed at the top of the results. See [U4.03.01].