Operator STAT_NON_LINE

1 Drank

Compute the mechanical evolution or coupled thermo-hydro-mechanics, into quasi-static, of a structure in 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 poursuite (the calculated 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 calculated 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 Syntax

```
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                ,                             [fonc.
/formule]
      TYPE_CHARGE=/          "FIXE_CSTE"                       [DEFAULT]
                             /"FIXE_PILO"
                             /"SUIV"
                             /"DIDI"
                          ),

◊  CONTACT=contact           ,                                  [char_contact]

◊SOUS_STRUC=_F               (
   ♦CAS_CHARGE=chi           ,                                  [char_meca]
                          ♦/TOUT=' OUI',
[DEFAULT]
      /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]
       |VARI=vain                ,                              [cham_elem]
       |DEPL=depl                ,                              [cham_no]
       |STRX=strx                                               [cham_elem]
      /EVOL_NOLI          =evol ,                               [evol_noli]
      /NUME_ORDRE         =nuini   ,                               [I]
      /INST               =instini ,                            [R]
    ◊  PRECISION=/1.0E-3             ,                             [DEFAULT]
                             /prec,                             [R]
   ◊CRITERE=/                   "RELATIF",                        [DEFAULT]
                              /"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          ,                          [DEFAULT]
                            /prec,                            [R]

                          ),

    ◊CRIT_QUALITE=_F            (
       ◊/ERRE_TEMPS_THM=/         "NON"                          [DEFAULT]
                          =     "OUI'
                          ),

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

    ◊  NEWTON=_F                  (
          "          TANGENT" PREDICTION=/,                     [DEFAULT]
                               /"ELASTIC",
                               /"EXTRAPOLATES",
                               /"DEPL_CALCULE",
          EVOL_NOLI=evol_noli           ,                      [evol_noli]
          "              TANGENT" MATRICE=/,                   [DEFAULT]
                               /"ELASTIC"
          REAC_INCR=/1                  ,                        [DEFAULT]
                               /mf,                             [I]
          REAC_ITER=/0                  ,                        [DEFAULT]
                               /it,                             [I]
           REAC_ITER_ELAS=/0            ,                        [DEFAULT]
                               /it,                             [I]
          PAS_MINI_ELAS=/0              ,                        [DEFAULT]
                               /pasmini,                        [R]
                          ),

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

    ◊  PILOTAGE=_F                (
        ♦  TYPE=/                    "DDL_IMPO",
                               /"SAUT_IMPO",
                               /"LONG_ARC",
                               /"SAUT_LONG_ARC"
                               /"ANA_LIM",
                               /"DEFORMATION",
                               /"PRED_ELAS",
                          TOUT=' OUI',                            [DEFAULT]
           /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            .,                               [DEFAULT]

                              /cmult,                             [R]
         ETA_PILO_R_MAX=etarmax     ,                             [R]
         ETA_PILO_R_MIN=etarmin     ,                             [R]
         ETA_PILO_MAX=etamax        ,                             [R]
         ETA_PILO_MIN=etamin                                         [R]
         PROJ_BORNES=/       "OUI'                               [DEFAULT]
                             /"NON"
         EVOL_PARA=/            "SANS"                           [DEFAULT]
                             /"DECREASING"
                             /"CROISSANT"
         SELECTION=/            "NORM_INCR_DEPL",                   [DEFAULT]
                             /"ANGL_INCR_DEPL",
                             /"residual",
                       ),

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

◊CONVERGENCE=_F            (
         /RESI_GLOB_RELA   =1.E-6   ,                                [DEFAULT]
          / | RESI_GLOB_MAXI=resmax    ,                            [R]
          | RESI_GLOB_RELA=resrel      ,                            [R]
          | RESI_COMP_RELA=rescmp      ,                            [R]
          | RESI_REFE_RELA=resref      ,                            [R]
         SIGM_REFE=sigref           ,                             [R]
         EPSI_REFE=sigref           ,                             [R]
         DEPL_REFE=depref           ,                             [R]
         FORC_REFE=forref           ,                             [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      ,                                [DEFAULT]
                             /maxelas,                           [I]
         ITER_GLOB_MAXI=/10      ,                               [DEFAULT]
                             /maglob,                            [I]
         ARRET=/                "OUI',                           [DEFAULT]
                              /"NON",
                       ),

◊    CRIT_STAB=_F           (
         TYPE=/              "FLAMBEMENT"                        [DEFAULT]
                         =    "STABILITY"
          NB_FREQ=/3            ,                                 [DEFAULT]
                              /nbfreq,                           [I]
         COEF_DIM_ESPACE=/5      ,                               [DEFAULT]
                             /coef,                             [I]
          RIGI_GEOM=/           "OUI',                          [DEFAULT]
                         =    "NON",
         MODI_RIGI=/            "NON",                          [DEFAULT]
                         =    "OUI',
         CHAR_CRIT=/             (- 10,10),                        [DEFAULT]
                             /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             ,                          [DEFAULT]
                             /prec,                           [R]
           ◊   CRITERE=/     "RELATIF",                       [DEFAULT]
                             /"ABSOLU",
           SIGNE=/              "POSITIF_NEGATIF",                 [DEFAULT]
                          =    "POSITIF",
                          =    "NEGATIF",
           PREC_INSTAB=/1.E-6        ,                          [DEFAULT]
                             /prec_instab,                    [R]
                           ),

      ◊ENERGIE=_F                ()

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

      ◊  AFFICHAGE=_F            (
            ◊  UNITE=/unite                                        [I]
            ◊  NOT            =/unite                                 [I]
            ◊  INFO_RESIDU=/     "NON",                        [DEFAULT]
                             /"OUI'
            ◊  INFO_TEMPS=/   "NON",                       [DEFAULT]
                             /"OUI'
                          ),
      ◊  OBSERVATION     =    _F (
           ◊TITER           =titer ,                          [list_k]
            ♦  NOM_CHAM=           |' DEPL',
                                  |' VITE',
                                  |' ACCE',
                                  |' DEPL_ABSOLU',
                                  |' VITE_ABSOLU',
                                 |' ACCE_ABSOLU',
                                 |' SIEF_ELGA',
                                 |' VARI_ELGA',
                                  |' FORC_NODA',
                                  |' VALE_CONT',
           ◊EVAL_CHAM=/           "VALE",                          [DEFAULT]
                                 /"MIN",
                                 /"MAX",
                                 /"MOY",
                                 /"MINI_ABS",
                                 /"MAXI_ABS",
           ♦  NOM_CMP=lnocmp             ,                          [l_Kn]
           ◊EVAL_CMP=/            "VALE",                          [DEFAULT]
                                 /"FORMULA",
           {If EVAL_CMP=' FORMULE'
              ◊FORMULE         =formes
[formule_aster]
```

```
          }
        {If CHAM is of type ELGA    (SIEF_ELGA, VARI_ELGA)
           ◊                TOUT=' OUI',                          [DEFAULT]
            ◊   /GROUP_MA        =lgrma   ,
[l_gr_maille]
            ◊   / MAILLE=lma       ,                             [l_maille]
           ◊   EVAL_ELGA=/          "VALE",                        [DEFAULT]
                                    /"MIN",
                                    /"MAX",
           {If EVAL_ELGA = ' VALE'
               ♦ POINT=pi                   ,                    [I]
               ◊ SOUS_POINT=spi             ,                    [I]
           }
        }
        {If CHAM is of type NOEU
           ◊                TOUT=' OUI',                          [DEFAULT]
            ◊   /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",                           [DEFAULT]
                            /"ABSOLU",
        { If             CRITERE=' RELATIF'
            ◊ PRECISION=/1.0E-6          ,                       [DEFAULT]
                              /prec,                             [R]
        }
        { If             CRITERE=' ABSOLU'
            ♦ PRECISION=prec          ,                          [R]
        }
    ),
◊   SUIVI_DDL=_F                (
        ◊TITER           =titer ,                                 [list_k]
         ♦   NOM_CHAM=            |' DEPL',
                                 |' VITE',
                                 |' ACCE',
                                |' SIEF_ELGA',
                                |' VARI_ELGA',
                                 |' FORC_NODA',
        ◊EVAL_CHAM       =    "VALE",                             [DEFAULT]
                              /"MIN",
                              /"MAX",
                              /"MOY",
                             / "MINI_ABS",
                              /"MAXI_ABS",
        ♦   NOM_CMP=lnocmp          ,                             [l_Kn]
        ◊EVAL_CMP=/             "VALE",                            [DEFAULT]
                              /"FORMULA",
        {If EVAL_CMP=' FORMULE'
           ◊FORMULE         =    forms                           [formule_aster]
         }
        {If CHAM is of type ELGA    (SIEF_ELGA, VARI_ELGA)
           ◊                TOUT=' OUI',                          [DEFAULT]
            ◊   /GROUP_MA        =lgrma   ,
[l_gr_maille]
            ◊   / MAILLE=lma       ,                             [l_maille]
```

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

3 Operands

3.1 Operands MODELS /CHAM_MATER /CARA_ELEM

♦MODELE = Mo

Name of the model whose elements are the object of mechanical computation.

♦CHAM_MATER = chmat

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

◊CARA_ELEM = carac

Name 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: so the model does not contain such elements, it is not useful; on the other hand, so the model contains such elements, it is compulsory.

3.2 Key word EXCIT

♦ EXCIT=_F ()

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

3.2.1 Operands CHARGE

♦ CHARGE = 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 local 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 local variables initial (associate at a field of initial temperature different from the reference temperature), the stress field and local 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 keys SIGM, DEPL, VARI in ETAT_INIT .* ***To avoid such situations which can lead to miscalculations, 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 to calculate the resultant of the forces, result is to be multiplied by 2π having the total resultant on the structure supplements. 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",        [DEFAULT]
                /"SUIV",
                /"DIDI",
                /"FIXE_PILO"
```

By default, TYPE_CHARGE 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 TYPE_CHARGE is worth "FIXE_PILO", the loading is always fixed (independent of the geometry) but will be controlled thanks to the key word CONTROL [§3.1222]. The loads controllable must result from operator AFFE_CHAR_MECA or AFFE_CHAR_MECA_F (if it is not a function depending on time) and not to 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.

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 or affected by FONC_MULT).

If TYPE_CHARGE is worth "SUIV", the loading is known as "follower", i.e. it depends on the value of the unknowns: for example, the pressure, being a loading applying in the normal direction to a structure, depends on the geometry brought up to date of this one, and thus on displacements. A following loading is revalued with each iteration of the algorithm of resolution.
Currently the loadings which can be described as follower are the loading of gravity for the element of CABLE_POULIE, the pressure for the 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).

Caution:

> *When a loading of pressure is following (TYPE_CHARGE=' SUIV') and that it depends on space (i.e. variables X, Y or Z in AFFE_CHAR_MECA_F) then the evaluating of the pressure will be done on the reactualized geometry. The variables X, Y or Z will be thus those of the current position of the point of evaluating of the pressure.*

If TYPE_CHARGE is worth "DIDI then " the conditions of Dirichlet (imposed displacements, linear conditions) will apply to the displacement increment from the time given under ETAT_INIT/NUMÉRIQUE_DIDI (by default moment of recovery of computation) and not to total displacement. For example for a displacement imposed (key word DDL_IMPO of operator

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 Key word SOUS_STRUC

◊SOUS_STRUC

This key word factor allows to specify which are the loadings to be used for under - static structures which then are 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 rest 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 incidence.

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 cause difficulties of convergence because of "the lapse of memory" of the state of preceding contact.

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

- So the model of computation 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;
- So the model given in the initial state is included in the model computation, Code_Aster starts by copying the values of the initial state for meshes common the then complete one with the value zero;

In the last case, it is necessary to take care of the meaning which can have a local variable null initial in computation.

3.6.1 Operands SIGM/VARI/DEPL/ STRX

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

sig is the stress field with Gauss points, vain is the field of the local variables to 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 of the command CREA_CHAMP, or be read in a file by the command LIRE_RESU.

3.6.2 Operands EVOL_NOLI

```
/ EVOL_NOLI = evol
```

Name 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 archivage NUME_ORDRE or of the time of archivage INST to carry out the poursuite of computation.
If NUME_ORDRE or INSTS are not filled, one takes the last existing number filed in evol.

```
◊  NUME_DIDI     = nudidi
```

In the case of loadings of the type Dirichlet differential ("DIDI"), one gives under NUME_DIDI the number of archivage of the mechanical state (displacements) which is used as reference for the application of these boundary conditions. By default 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), associated time ago.
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 Example (behavior by default)

```
LIST1   = DEFI_LIST_REEL (debut =0.,
                          INTERVALLE =_F (UNTIL = 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 - Example to show the interest of INST_ETAT_INIT (two different lists of times)

```
LIST1   = DEFI_LIST_REEL (debut =0.,
                          INTERVALLE =_F (UNTIL = 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 - Example to show the interest of INST_ETAT_INIT (practical when the cyclic one is made)

```
LIST1   = DEFI_LIST_REEL (debut =0.,
                          INTERVALLE =_F (UNTIL = 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 accuracy/CRITERE

◊ accuracy = prec

Confer [U4.71.00] for detailed syntax
This parameter is used to locate the good sequence number (NUME_ORDRE) when the user informs time (INST). Indeed, times in STAT_NON_LINE are located 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
INSTS	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
INSTS	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 default ($1\text{E-}6$). 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 the parameter accuracy to be able to select time (in the example, PRECISION=1E-8 will be appropriate).

3.7 Key word INCREMENT

♦ INCREMENT=_F ()

Defines the 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]

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

♦ If LIST_INST = litps [list_inst]

times of computation are those defined in the concept litps by the 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 keys INST_INIT or NUME_INST_INIT, the default 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 archivage (see keyword ARCHIVAGE), initial time in poursuite is the last step filed and not that defined in INST_INIT.

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

final moment (last calculated step) is indicated same way 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 time step is activated, NUME_INST_FIN does not take account and always works of it on the list of times initial. NUME_INST_INIT and NUME_INST_FIN are active only with the initialization.

A - simple Example (behavior by default)

```
LISTE = DEFI_LIST_REEL (   debut =0.,
                        INTERVALLE =_F (UNTIL = 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 - Example to show the interest of INST_INIT

```
LISTE = DEFI_LIST_REEL (   debut =0.,
                        INTERVALLE = _F (UNTIL = 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 accuracy

◊ accuracy = prec

cf [U4.71.00] for detailed syntax

This parameter is used to locate good sequence number (NUMÉRIQUE_INST_FIN/NUMÉRIQUE_INST_INIT) when the user informs time (INST_FIN/INST_INIT). Indeed, times in STAT_NON_LINE are located 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
INSTS	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
INSTS	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 default ($1\text{E-}6$). 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 the parameter accuracy to be able to select time (in the example, PRECISION=1E-8 will be appropriate).

3.8 Operand CRIT_QUALITE

◊ **ERRE_TEMPS_THM=/ "NON" [DEFAULT]**
/"OUI'

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

3.9 Operand METHODE

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

Makes it possible to choose the method of resolution of the nonlinear incremental problem.

/"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 step of loading (see [R5.03.01]).

3.10 The purpose of key word NEWTON

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

3.10.1 Operand PREDICTION

◊ **PREDICTION = "TANGENT"**
/"ELASTIC"
/"EXTRAPOLATES"
/"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=' ELASTIQUE' were chosen.

/"TANGENT"

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

/"ELASTIC"

the elastic matrix is used (option RIGI_MECA).

/One "EXTRAPOLATES

" time step calculates the estimate of the displacement increment from the total increment obtained like solution with preceding (balanced by the ratio of time step). 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 calculated. 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 nombre of iterations necessary can be high. The use of extrapolation can improve the performances. /"

DEPL_CALCULE" Makes it possible

to propose like displacement for the prediction with each time step, the displacement given by a mechanical history specified under key word EVOL_NOLI (§ 14)14Displacement is project on kinematically admissible all the fields, as for the method EXTRAPOLATES . Note:

- The methods "EXTRAPOLATES " 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; It
- is necessary that the displacements used in "EXTRAPOLATES " and "DEPL_CALCULE " result from computation using the same mesh because the limiting conditions must be coherent; Because of
- this projection on the limiting conditions, these two options are incompatible with the functionalities of CONTROL . Attention

, because of impossibility "of correctly projecting" the limiting conditions from one mesh to another, it from now on is strongly disadvised using "EXTRAPOLATES " and "DEPL_CALCULE " from a mesh different from current computation. Utility

: If

- one obtained a first solution on the same mesh with others materials parameters or another behavior, the fields of displacements can be re-used in computation. That
- makes it possible to reduce the core memory and to preserve these results for a later poursuite. For a large computation, one can store only displacements at all times with the formats IDEAS or MED in IMPR_RESU . If one wants to recompute the stresses and local variables, one makes a LIRA _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 (to avoid the Dirichlet loadings which impose a linear resolution) as well as a convergence criterion, even if this information is not taken into account. Operand

3.10.2 MATRICE ◊

```
MATRICE     = /    "TANGENT " ◊
REAC_INCR      =/1 [ DEFAULT                    ] /mf
                  ◊
REAC_ITER      =/0 [ DEFAULT                    ] /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 time step. ◊

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

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 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 a maximum nombre of iterations specific to the use of the matrix elastic and different from that associated with the use of the tangent matrix. One can
with the key word define a frequency of reactualization of the matrix of discharge REAC_ITER_ELAS (similar 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), to 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 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. ◊ MATRICE

```
=/"ELASTIC       " 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 means of the modulus Young given under key word ELAS of operator DEFI_ MATERIAU, and not the slope at the origin of the curve of tension 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]). Operand

3.10.3 EVOL_NOLI ◊ **EVOL_NOLI**

= evol_noli Name of

the concept of the type evol_ noli which will be useful in the prediction by DEPL_CALCULE . Key word

3.11 RECH_LINEAIRE ◊RECHERCHE

_LINEAIRE=_F () the 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 the linear search with strains GROT_GDEP for modelizations COQUE_3D and in the presence of contact. Operand

3.11.1 METHODE ◊METHODE

```
= "CORDE  " [DEFAULT                    ]/"MIXTE
          "/"CONTROL
          " Makes it possible
```

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]). Method CORDE (by default) 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). The method

CONTROL is reserved for control of the type DEFORMATION , PRED_ELAS and LONG_ARC (see §22). It22 the only method usable with this kind of control. For the control of the type DDL_IMPO , one can use CORDE or MIXTE . Operand

3.11.2 RESI_LINE_RELA /ITER_LINE_MAXI ◊RESI

```
_LINE_RELA    =/1.E-1   [DEFAULT          ] /reslin
                    ◊ ITER_LINE_MAXI
    =/3 [DEFAULT                ] /itelin
                    They are
```

the parameters of the linear search. One gives the maximum nombre of iterations itelin to be carried out and the accuracy reslin to reach to carry out the convergence of the linear search. It is advised not to use the linear search with of the contact. For

method CORDE , It is not necessary to specify an accuracy nor a nombre of iterations 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 method CORDE . On the other hand
, for method MIXTE , on problems with damage, several tens of iterations are often effective. Operands

3.11.3 RHO_MIN /RHO_MAX/RHO_EXCL ◊RHO

```
_MIN=/1.E-2          [ DEFAULT          ]/rmin
                [  R] ◊RHO
_MAX=/1.E+1          [ DEFAULT          ]/rmax
                [  R] ◊RHO
_EXCL=/9.E-3         [ DEFAULT          ]/rexc
                [  R] These          key words
```

fix the interval in I which one calculates the coefficient RHO of the linear search, in the form: [R5.03 $I=[rmin, rmax]-[-rexc, rexc]$ .01]. Key word

3.12 CONTROL ◊ CONTROL

=_F () When

the intensity of η 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), the key word CONTROL allows to control this loading via a node (or groups node) 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. The key word
• CONTROL is prohibited with the contact (except in case of the contact XFEM) . It
•is not possible to make CONTROL with PREDICTION = ' DEPL_CALCULE' or PREDICTION = ' EXTRAPOLE' (see § 19) 19

3.12.1 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

" Makes it possible

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

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

" Takes again

the principle of DDL_IMPO but to control the increment of the jump of displacement between the lips of a crack X-FEM. Only one direction is possible i , 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 of P_a the interface with the edges of the mesh a . This group describes all crack if GROUP_NO is not indicated (behavior by default), and only one part if it is it. Attention, this kind of control can be used only in modelization X-FEM. /"LONG_ARC

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

" Makes it possible

referenced by to control the intensity of η the loading 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: with

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

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

" Takes again

the principle of LONG_ARC but to control the norm of the increment of the jump of displacement between the lips of a crack X-FEM. One controls this norm on average on a set of points of intersection of P_a the interface with the edges of the mesh a . This group describes all crack if GROUP_NO is not indicated (behavior by default), and only one part if it is it. with

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

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 indicates $\boldsymbol{F}$ the controlled assembled loading, TYPE_ CHARGE=' FIXE_PILO', then the function of control is written simply: Except

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

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 Gauss point of structure sees its strain evolving in a monotonous way. The relation is checked: This mode

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

of control is valid for all the constitutive laws including in large deformations SIMO_MIEHE . /"PRED_ELAS

" PRED_ELAS

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

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

of control is valid only for models ENDO_FRAGILE (with the local version and version nonlocal GRAD_ EPSI), ENDO_SCALAIRE (with the non local 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 is difficult cmult to define first blow because the notion of output of criterion is not $\frac{\Delta t}{\text{cmult}}$ 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 is used $\frac{\Delta t}{\text{cmult}}$, 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 the key word CONTROL to start the problem and to obtain an initial state different ε^- from zero (if not division by zero for control by increment of strain). One carries out after a recovery 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. This is why ε^-, 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 the key word CONTROL in order to obtain an initial compactness of tension ε^- or. One carries out then a second STAT_NON_LINE in poursuite from the preceding initial state with the key word CONTROL . Note:

DEFORMATION

and PRED_ELAS are not available for the structural elements. Operands

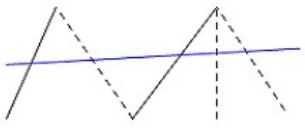
3.12.2 NOEUD /GROUP_NO/NOEUD

= 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

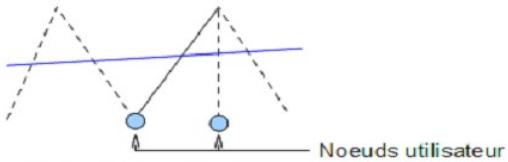
"SAUT_IMPO " and "SAUT_LONG_ARC ", the definition is more subtle since in modelization X-FEM one does not control the values on nodes but on points of intersection between the edges of the mesh and crack. In the continuation, one simply indicates by "edges" the intersected edges. The algorithm starts by building a set of independent edges which covers all crack (see fig.3.12.2-13.12.2-1, 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 3.12.2-2
- 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 3.12.2-4 an error is returned. Figure



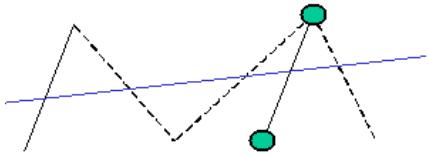
Sélection automatique

3.12.2 3.12.2-1 nonwell informed Figure

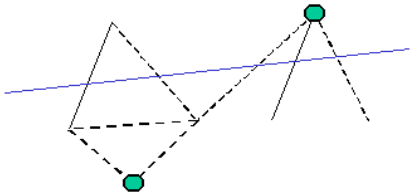


Sélection utilisateur

3.12.2 3.12.2-2 ends of nonindependent edges Figure



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



3.12.2 3.12.2-4 for node off-line to an intersected edge Table

3.1. 3.1

3.12.3 ANY MAILLE/GROUP_MA/TOUT

= "OUI' [DEFAULT]/GROUP _MA=lgrma /MESH =lma One gives

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. Operand

3.12.4 NOM_CMP ◊ NOM_CMP

= nomcmp It is

the name of the component (corresponding to the degree of freedom) used i for control ("DX" for example). To use only with "DDL_IMPO " or" LONG _ARC'. Operand

3.12.5 DIRE_PILO ◊ DIRE_PILO

= direpilo It is

the name of the direction according to i 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), "DTAN X2" 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 ". Operand

3.12.6 FISSURES ◊ CRACK

= fiss Name 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 ". Operand

3.12.7 COEF_MULT ◊ COEF_MULT

= cmult It is

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

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

       DEFI_LISTE_REEL (DEBUT=0., INTERVALLE
                        =_F (UNTIL    = 10, NOMBRE  =10), RESU=

       STAT_NON_LINE (EXCIT=_F (CHARGE             = PRESSION      ,
TYPE_CHARGE                                                 = '
FIXE_PILO'), CONTROL
                        =_F (TYPE         = "DDL_IMPO       ", NOEUD
                           = 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. (Here $P_{limite}=\eta\times P_x$ 1 P_x thus one is worth has the Yield-point load directly). If one on the structure imposes a pressure close P 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 . Operand

3.12.8 ETA_PILO_R_MAX /ETA_PILO_R_MIN ◊ ETA

```
_PILO  _R_MAX=etarmax      , [R] ◊ETA
_PILO_R_MIN=etarmin        , [R] These              two
```

key words make it possible to define the interval of search 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 for $-\infty$ 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. Operand

3.12.9 ETA_PILO _MAX/ETA_PILO_MIN ◊ ETA

```
_PILO   _MAX=etamax        , [R] ◊                  ETA
_PILO   _MIN=etamin        , [R] These                 two
```

key words makes 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 interval of search 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 at $etamin$ a low value, that makes it possible to stop computation before a tear/complete damage of the sample and to avoid thus the divergence with the last time step. The other possible use is that of as $etamax$ a maximum Yield-point load. Caution:

With

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

3.12.10PROJ_BORNES ◊ PROJ

```
_    BORNES=/"OUI'        [DEFAULT                    ]/"NON
                        " In the event of
```

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', one does not modify the values of eta, 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 control 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 de facto authorizes 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. Operand

3.12.11SELECTION ◊ /SELECTION

```
= "      NORM_INCR_DEPL", [DEFAULT     ]/"ANGL_INCR_DEPL
               ",/"residual
               "/"MIXTE
               " This 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 time step considered. "ANGL_INCR_DEPL
" makes it possible to select the value of control by the smallest angle between the displacement obtained for time step running and the displacement obtained for the time step preceding one. "residual
" 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 leaning on several strategies. One starts initially with strategy "NORM_INCR_DEPL" above . If the results of the function purpose (the norm of the displacement increment) are too close, one rocks for this iteration on the strategy "residual". 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 time step flow 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 the keyword SELECTION = ' ANGL_INCR_DEPL', it is important to keep in mind that this criterion requires two the time step *preceding ones. It will thus be necessary well to take care to correctly file the results of the preceding computation to the risk getting false results. An alarm informs the user. Operand*

3.12.12EVOL_PARA ◊ EVOL

```
_    PARA=/"SANS            " [DEFAULT                          ]/"DECREASING
                            "/"CROISSANT
                             " This operand
```

makes it possible to impose it to it growth or the decrease of the parameter of control. Key word

3.13 solver ◊ solver

=_F () the syntax

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

3.14 CONVERGENCE ◊ CONVERGENCE

=_F () If none

of the two operands following is present, then all happens like if: RESI_GLOB_RELA = 1.E-6. Operand

3.14.1 RESI_ GLOB_RELA/RESI_GLOB_MAXI ◊ |

RESI_GLOB_RELA =resrel , [R] the algorithm

continues the total iterations as long as: where

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

is $\boldsymbol{F}^n$ the residue of the iteration and n the vector $\boldsymbol{L}$ 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 is null $\boldsymbol{L}$ (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 becomes again $\boldsymbol{L}$ different from zero, one passes by again automatically with relative convergence criterion RESI_GLOB_RELA . However , this mechanism of swing 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 step 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, to 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: where

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

is $\boldsymbol{F}^n$ the residue of the iteration (cf n [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 evaluating of the residue of equilibrium. What does not cause of results false but when the loading becomes null, i.e. when is null L (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 becomes again L different from zero, one passes by again automatically with relative convergence criterion RESI_GLOB_RELA. Operand

3.14.2 RESI_COMP_RELA ◊ |

```
RESI_COMP_RELA    =rescmp  , [R] This                    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 (for example cmp in THM,). One $cmp=\left(DX, DY, DZ, PRE1, PRE2, TEMP\right)$ normalizes then these under-vectors by the corresponding internal force. Thus, the algorithm continues the total iterations as long as: where

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

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

at time n are calculated at the beginning of time step from result resulting from time step preceding. 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 is interested only in problems of the evolutionary type (THM) where resident of strong contrasts between the various unknowns. Operand

3.14.3 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] This             operand
```

results in estimating the convergence of the algorithm of Newton in the following way. From a reference, which can be: A stress

- sigref ; A strain
- epsref for incompressible elements, elements of grid and membrane; A local variable
- varref if one uses nonlocal models with deformation gradient; A heat flux
- fthref in a case THM; Two
- hydrous flux fh1ref and fh 2ref 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 calculated (of a the same F^{ref} vector length than the vector residue). Convergence will be carried out if and only if: Operand

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

3.14.4 ITER_GLOB_MAXI ◊ ITER_GLOB_MAXI

=/10 [DEFAULT] maximum
/maglob

Nombre of iterations carried out to solve the total problem at every moment (10 per default). This test is always carried out except in the case of the recutting of time step by the method “EXTRAPOLATES ″. 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 the formulation CONTINUE in generalized Newton, it is often necessary to increase the nombre of iterations of Newton. Operand

3.14.5 ITER_GLOB_ELAS ◊ ITER_GLOB_ELAS

=/25 [DEFAULT] maximum
/maxelas

Nombre of iterations carried out with the elastic matrix when one uses key word PAS_MINI_ELAS of the key word factor NEWTON (see §19)19 to solve the total problem at every moment (25 per default). It is pointed out

that PAS_MINI_ELAS makes it possible to pass from the tangent matrix to the elastic matrix when 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. Operand

3.14.6 ARRET ◊ARRET

=/“OUI' [DEFAULT] 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” [DEFAULT] 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. Key word

3.15 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 (TYPE= “FLAMBEMENT ”): This criterion

is then calculated in the following way: at the end of one time step, in small disturbances, one solves. is $det\left(K^T-\lambda\cdot K^g\right)=0$ K^T the coherent tangent matrix at this time. is K^g the geometrical stiffness matrix, calculated starting from the stress field at this time. In practice
, the loading is unstable if (in fact $|\lambda|<1$). One $-1<\lambda<0$ calculates 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 deformations, one solves because $det\left(K^T-\lambda\cdot I\right)=0$ contains K^T then. K^g The criterion is then a criterion of instability: when changes λ 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 object result, under name MODE_FLAMB . This eigen mode can be extracted and visualized (like a field of displacements or a classical 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 a dissipative mechanics (TYPE = "STABILITY") :

To treat this typical case, one forces to take as geometrical stiffness matrix the matrix identity. One $K^g=I_d$ searches then the minimum of following quadratic functional calculus: where

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

3.15.1 LIST_INST /INSTS /PAS_CALC ◊/"

```
LISTE_INST'=  liste_r8/"INST
"= l_r8      /"NOT
_CALC'= npas  times
```

for which one wants to do a calculation of stability are given by a list of times (list_r8 or L _r8) oupar a frequency PAS_CALC (all of time $npas$). In the absence of
these keywords the criterion is calculated with all time step. Operand

3.15.2 accuracy /CRITERE ◊ accuracy

```
=/1.e-6       [DEFAULT                ] /prec
             ◊ CRITERE
= "RELATIF      ", [DEFAULT           ]/"ABSOLU
               ", Allows
```

to select times, confer *[U4.71* .00] Operand

3.15.3 NB_FREQ ◊ NB_FREQ

```
= 3   , [ DEFAULT                     ]/nbfreq
          key word
```

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

3.15.4 COEF_DIM_ESPACE ◊ COEF_DIM_ESPACE

```
= 5, [DEFAULT                    ]/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 . Operand

3.15.5 RIGI_GEOM ◊ RIGI_GEOM

```
= "OUI'   , [DEFAULT                ]/"NON
            " key word
```

RIGI_GEOM ("OUI' by default) 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 deformations). To choose "NON" means that the geometrical stiffness matrix is replaced by the identity. Operand

3.15.6 MODI_ RIGI ◊ MODI

```
_   RIGI = "NON " , [DEFAULT                ]/"OUI'
                key word
```

MODI_ RIGI ("NON " by default) 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. Operand

3.15.7 CHAR_CRIT ◊ TANK

```
_   CRIT=/(- 10,10     ), [DEFAULT            ] /intcc
                key word
```

CHAR_CRIT makes it possible to save time by making only one test of Sturm type in the provided bande de fréquence. If at least a frequency is found, then one calculates really the values of the critical loads in this interval. This key word can be used only under the condition RIGI_GEOM = "OUI' (default value of option RIGI_GEOM). Operand

3.15.8 DDL_EXCLUS ◊ DDL_EXCLUS

```
= ("DX "  , "DY",…) Key word
```

DDL_EXCLUS (empty list by default) 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 Lagrangian and of negative eigenvalues. In

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 taking account of the excluded degrees of freedom. For example, one must use this option for the models fluid-structure

coupled (formulation, confer (u, p, ϕ) *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]. Operand

3.15.9 DDL_STAB ◊ DDL

_ STAB = ("DAMG ",…) Key word

DDL _ 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. Operand

3.15.10 SIGNE ◊ SIGNE

= "POSITIF _NEGATIF", [DEFAULT] = "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 declaration 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 default 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 criterion SIGNE with the key word . It is pointed out that the critical load calculated 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. Operand

3.15.11 PREC_ INSTAB ◊ PREC

_ INSTAB=/1.E-6 , [DEFAULT] /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 the key word preceding SIGNE . Key word

3.16 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_ COUNTS [U4.71 .02]. Key word

3.17 ARCHIVAGE ◊ ARCHIVAGE

=_F () Makes it possible

to file or certain results at all or certain times of computation. In the absence of
this key word all time step are filed, including times of computations lately created by automatic recutting of time step. The archivage makes it possible saved to reduce appreciably the size of the bases by selecting times. Operand

3.17.1 LIST_INST /INST /PAS_ARCH ◊/"

LISTE_INST'= liste_r8/"INST
"= l_r8 /"NOT
_ARCH'= npas the designation

of times to be stored is carried out either by a list of times (list_r8 or L _r8) or then by a frequency of archivage (all npas of time). In the absence of
these keywords all time step are filed. Two

note: the last

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

3.17.2 accuracy /CRITERE ◊ accuracy

= /1.e-6 [DEFAULT] /prec
◊ CRITERE
= "RELATIF ", [DEFAULT]/"ABSOLU
", cf [

U4.71.00] Operand

3.17.3 CHAM_EXCLU ◊ CHAM_EXCLU

Makes it possible

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

3.18 DISPLAY ◊ DISPLAY

=_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". Operand

3.18.1 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). Operand

3.18.2 NOT ◊ NOT

= not Frequency

of reactualization of the display in the message file. This operand makes it possible to reduce the volume of the printings, particularly in explicit dynamics. He does not affect the reactualization of the file to the format .csv (keyword UNITE). Operand

3.18.3 INFO_RESIDU ◊ INFO_RESIDU

= "NON" , [DEFAULT]/"OUI'
This 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. Operand

3.18.4 INFO_TEMPS ◊ INFO_TEMPS

= "NON " , [DEFAULT]/"OUI'
This operand

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

3.19 OBSERVATION ◊ OBSERVATION

=_F () ◊ TITER
= title This key word

makes it possible post-to treat certain fields at nodes or with the elements on parts of model times D" a list (known as D" observation) more refined generally than the list of the times filed defined in the key word ARCHIVAGE [§3.17]34 one the model stores all the fields on all). 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, to calculate the norm of the stresses, within the meaning of Von-Put, and to store it in the array of observation. This key word

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

3.19.1 LIST_INST /INSTS /PAS_OBSE ◊/

"**LISTE** _INST'= liste_r8/"INST
"= l_r8 /"NOT
_OBSE'= npas These operands

make it possible to define in the choices a list D" urgent D" observation. LIST_INST , INSTS and PAS_OBSE have the same meaning as of the same operands name being used to define a list D" archivage. PAS_OBSE playing the same part as PAS_ARCH in ARCHIVAGE [§3.17].34

3.19.2 accuracy /CRITERE ◊ accuracy

```
= 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 L" observation. Operands

3.19.3 NOM_CHAM /NOM_CMP ♦ NOM_CHAM

```
= nomcham     ♦ NOM_CMP
= nomcmp      These 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 . Operands

3.19.4 ANY NOEUD/GROUP_NOEUD/MAILLE/GROUP_MA ◊ /TOUT

```
= "NON        "   [DEFAULT                              ] "/NOEUD
                   OUI'
= No       [No    ]                                  /GROUP_NO
=grno      [grno                                ] /MAILLE
=lma [my           ] /GROUP_MA
=lgrma      [  grma]                                 These operands
```

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

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

3.19.5 of a field ELGA ◊ EVAL_CMP

```
=/"   VALE    ", [DEFAULT                                ]/"FORMULA
               " ◊ FORMULATES
= forms       [formula                                   _aster] One starts
```

by choosing the components or the formula between the components: If EVAL_CMP
- =" VALE ", one extracts simply the list of the components given by NOM_CMP . If EVAL_CMP
- = "FORMULA ", one evaluates the formula given by the key-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    ", [DEFAULT                          ]/" MIN
                      " ,/" MAX
                       " , ♦ /POINT
=pi [I]                  ◊                                /SOUS_
POINT =spi [     I   ] Once
```

evaluated the components or the formula on the components, one can: To 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). To 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 will have as many achievements as of points requested, multiplied by the number of components 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      ", [DEFAULT                          ]/"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 Once evaluated the components (or formulates it on the components), have nsi that the point/subpoint of extraction, one can: Meshes

- to extract these values on all with EVAL_ CHAM= " VALE ". To ask
- to extract maximum EVAL_ CHAM= "MAX" , minimum EVAL_ CHAM= "MIN" or average EVAL_ CHAM= "MOY" . Example:

To extract L E maximum from the trace of the tensor of the stresses on GROUP _MA=' LOUSE" traces

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

3.19.6 of a field NOEU ◊ EVAL_CMP

```
=/"   VALE    ", [DEFAULT                          ]/"FORMULA
               " ◊ FORMULATES
= forms       [formula                             _aster] One starts
```

by choosing the components or the formula between the components: If EVAL_CMP

- = "VALE ", one extracts simply the list of the components given by NOM_CMP . If EVAL_CMP
- = "FORMULA ", one evaluates the formula given by the key-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    ", [DEFAULT                          ]/"MIN
                " ,/" MAX
                " ,/" MOY
                " , Once
```

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

- to extract these values on all with EVAL_ CHAM= "VALE ". To ask
- to extract maximum EVAL_ CHAM= "MAX" , minimum EVAL_ CHAM= "MIN" or average EVAL_ CHAM= "MOY" . Example:

To extract the maximum of component DX from displacement on GROUP_NO = ' LOUSE "OBSERVATION

```
=_F (NOM_CHAM     =" DEPL ", GROUP_NO
                    = "LOUSE ", EVAL_CHAM
                    = "MAX", NOM_CMP
                    = (" DX ", ), ) Contained
```

3.19.7 array the array

will contain to the maximum 16 columns. NOM_OBSERVATION

K80	Name given	automatically or by key key TITER TYPE_
OBJET K16	the array	thus contains only actual values R NOM _SD
K24	" "	NUMÉRIQUE_
REUSE I Index		of re-use of the array in the event of REUSE NUME_
OBSE I Sequence number		of the observation INST
R Urgent		of observation NOM_CHAM
K16	Name of	the field observed EVAL_CHAM
K8 Type		of evaluating of field NOM_CMP
K8 Name	of	component observed EVAL_CMP
K8 Type		of evaluating of the component NOEUD
K8 Node		where the observation (field at nodes) is carried out NETS
K8 Nets		where the observation (field with meshes) is carried out EVAL_ELGA
K8 Type		of evaluating of the fields to Gauss points POINT
I Not		integration where the observation (fields with meshes) is carried out SOUS_POINT
I Subpoint		of integration where the observation (fields with meshes) is carried out VALE
R Value		parameter

NUME_ REUSE is useful in the event of enrichment of data structure result. Indeed, if the recovery result crushes old sequence numbers in data structure (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. Key word

3.20 SUIVI_DDL ◊ SUIVI_DDL

```
=_F () This key word
```

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

SUIVI_DDL has same syntax as OBSERVATION for the extraction of the fields, except that one does not give an information about times to be extracted, since one carries it out with each iteration of Newton (there are not key keys LIST_ INST/INST/PAS_OBSE/CRITERE/PRECISION). ◊ TITER

=ltitre , [list _ K] This key word

expects a list of three character strings to the maximum and makes it possible to name the column of the display board. The character strings are truncated with 16 characters. Contained

3.21 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 (with Gauss points) of the stresses; VARI_ELGA
- : field (with Gauss points) of the local 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 (with Gauss points) of the statutes of contact for case XFEM with contact; COHE_
- ELGA: field (with Gauss points) of the parameter of cohesion for case XFEM with RELATION = ' CZM"; SECO _
- ELGA: field (with 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 Keyword

	Standard origin	Description	INST
Value		of the time of computation R EXCIT	
EXCIT	Information	on the loadings K24 MODELS	
MODELS	Models	elementary K8	
CARAELEM	CARA_ELEM	Characteristics R K8	
CHAMPMAT	CHAM_MATER	Material field K8	
PARM_THETA	COMP_ INCR/PARM_THETA	Parameter of integration of constitutive law ITER	_
GLOB Nombre total		of iterations of Newton I TANK	_
MINI minimum		Loading reached during time step R ETA	_PIL OTA GE
Parameter		of control R When	

one searches 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 Keyword

	Standard origin	Description	CHAR_C RIT
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" Value	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 Operand

3.22 INFO ◊ INFO

=inf Makes it possible

to carry out in the message file various intermediate printings. Other printings are done systematically during the nonlinear calculation, independently of value affected to the 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. Operand

3.23 TITER ◊ TITER

= tx tx is

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