

Operator DEFI_LIST_INST

1 Drank

To define list of computation moments, like his management for the iterative algorithms of resolution (commands `STAT_NON_LINE` and `DYNA_NON_LINE`):

- possibilities of recutting of time step in the event of failure,
- adaptability of time step.

Product a data structure `list_inst`.

Contents

1 But1.....	
2 Syntaxe3.....	
3 Opérandes6.....	
3.1 Key word DEFI_LIST6.....	
3.1.1 Operand METHODE6.....	
3.1.2 Operands LIST_INST and VALE6.....	
3.1.3 Operands PAS_MINI, PAS_MAXI, NB_PAS_MAXI6.....	
3.2 Key word ECHEC7.....	
3.2.1 Definition of the causes of échec7.....	
3.2.1.1 Event "ERREUR"	7
3.2.1.2 Event "DIVE_RESI"	8
3.2.1.3 Event "DELTA_Grandeur"	8
3.2.1.4 Event "COLLISION"	9
3.2.1.5 Event "INTERPENETRATION"	9
3.2.1.6 Event "INSTABILITY"	
3.2.2 Definition of the actions to carry out in the event of échec9.....	
3.2.2.1 Action "ARRET"	10
3.2.2.2 Action "DECOUPE"	10
3.2.2.3 Action "AUTRE_PILOTAGE"	13
3.2.2.4 Action "ITER_SUPPL"	13
3.2.2.5 Action "ADAPT_COEF_PENA"	13
3.2.2.6 Action "CONTINUE"	14.3.3
Key word ADAPTATION14.....	
3.3.1 Operand EVENEMENT14.....	
3.3.2 Operands NB_INCR_SEUIL, NOM_PARA, CRIT_COMP, VALE_I15.....	
3.3.3 Operand MODE_CALCUL_TPLUS15.....	
3.3.4 Operand PCENT_AUGM16.....	
3.3.5 Operands VALE_REF, NOM_CHAM, NOM_CMP16.....	
3.3.6 Operand NB_ITER_NEWTON_REF16.....	
3.4 Operand INFO16.....	

2 Syntax

```
deflist [list_inst] =DEFI_LISTE_INST (
```

1) key words for the definition a priori of the list of time

```
DEFI_LISTE=_F (
    "          MANUAL" ◆METHODE=/, [DEFAULT]
                        /"CAR",
    ◆/LISTE_INST=LIST , [l_r8]
      /VALE          =VALE , [R]
      /◆RESULTAT=RESU , [result]
        ◆SUBD_PAS=NPAS [I]

    # if METHODE = "AUTO"
      ◆ PAS_MINI      = pasmin, [R]
      ◆ PAS_MAXI      = pasmax, [R]
      ◆NB_PAS_MAXI= /1000000 [DEFAULT]
                        /nbpasmax , [I]
    )
```

2) key words for the behavior in the event of failure

```
◆ ECHEC=_F ( [DEFAULT]
  ◆EVENEMENT=/ "ERREUR",
                /"DIVE_RESI",
                /"DELTA_Grandeur"
                /"COLLISION"
                /"INTERPENETRATION"
                /"INSTABILITY"

  # If EVENEMENT = "DELTA_Grandeur"
  ◆VALE_REF=VALREF , [R]
    ◆ NOM_CHAM= /"DEPL" ,
                /"SIEF_ELGA",
                /"VARI_ELGA",

  ◆NOM_CMP=CMP , [TXT]
  ◆ACTION=/ "DECOUPE", [DEFAULT]
            /"ARRET",

    # If ACTION = "DECOUPE"
    ◆SUBD_METHODE=/ "MANUEL", [DEFAULT]
                  /"AUTO",

      # If SUBD_METHODE = "MANUEL"
      ◆SUBD_PAS=4 , [DEFAULT]
        not, [R]

      ◆SUBD_NIVEAU=3 , [DEFAULT]
        niv, [I]

      ◆SUBD_PAS_MINI=0 , [DEFAULT]
        pasmin, [R]

      # If SUBD_METHODE= " AUTO "
      ◆SUBD_PAS_MINI=0 , [DEFAULT]
        pasmin, [R]

  # If EVENEMENT = "ERREUR"
  ◆ACTION=/ "DECOUPE", [DEFAULT]
            /"ARRET",
```

Warning : The translation process used on this website is a "Machine Translation". It may be imprecise and inaccurate in whole or in part and is provided as a convenience.

```
        /"REAC_PRECOND",
        /"ITER_SUPPL",
        /"AUTRE_PILOTAGE",

# If ACTION = "DECOUPE"/"REAC_PRECOND"/"AUTRE_PILTOAGE"
◇SUBD_METHODE=/          "MANUEL",          [DEFAULT]
                        /"AUTO",
# If SUBD_METHODE = "MANUEL"
◇ SUBD_PAS=4              . ,                [DEFAULT]
                        not,                [R]
◇SUBD_NIVEAU=3           ,                  [DEFAULT]
                        niv,                [I]
◇ SUBD_PAS_MINI=0        . ,                [DEFAULT]
                        pasmin,            [R]
# If SUBD_METHODE = " AUTO "
◇SUBD_PAS_MINI=0         ,                  [DEFAULT]
                        pasmin,            [R]

# If ACTION = "ITER_SUPPL "
◇ PCENT_ITER_PLUS=50     ,                  [DEFAULT]
                        pcent,            [I]
◇SUBD_METHODE=/          "MANUEL",          [DEFAULT]
                        /"AUTO",
# If SUBD_METHODE = "MANUEL"
◇SUBD_PAS=4              . ,                [DEFAULT]
                        not,                [R]
◇SUBD_NIVEAU=3           ,                  [DEFAULT]
                        niv,                [I]
◇SUBD_PAS_MINI=0        . ,                [DEFAULT]
                        pasmin,            [R]
# If SUBD_METHODE = " AUTO "
◇SUBD_PAS_MINI=0         ,                  [DEFAULT]
                        pasmin,            [R]

# If EVENEMENT = " COLLISION "
◇ACTION=/                "DECOUPE",        [DEFAULT]
                        /"ARRET",
# If ACTION = "DECOUPE"
◇SUBD_METHODE=/          "AUTO",          [DEFAULT]
                        /"MANUEL",
# If SUBD_METHODE = "MANUEL"
◇SUBD_PAS=4              . ,                [DEFAULT]
                        not,                [R]
◇SUBD_NIVEAU=3           ,                  [DEFAULT]
                        niv,                [I]
◇ SUBD_PAS_MINI=0        . ,                [DEFAULT]
                        pasmin,            [R]
# If SUBD_METHODE = " AUTO "
    ◆SUBD_INST=SUBD_INST      ,            [R]
    ◆SUBD_DUREE=SUBD_DUREE    ,            [R]

# If EVENEMENT = " INTERPENETRATION "
    ◆ PENE_MAXI=              penmax ,      [R]
◇ACTION=/                " ADAPT_COEF_PENA ", [DEFAULT]
                        /"ARRET",
# If ACTION = " ADAPT_COEF_PENA "
```

```
◇COEF_MAXI=1E12 [DEFAULT]
coemax , [R]

# If EVENEMENT = " DIVE_RESI " [R]
◇ACTION=/ " DECOUPE " , [DEFAULT]
◇SUBD_METHODE=/ "MANUEL" , [DEFAULT]
/ "AUTO" ,

# If SUBD_METHODE = "MANUEL"
◇SUBD_PAS=4 . , [DEFAULT]
not , [R]
◇SUBD_NIVEAU=3 , [DEFAULT]
niv , [I]
◇SUBD_PAS_MINI=0 . , [DEFAULT]
pasmin , [R]

# If SUBD_METHODE = " AUTO "
◇SUBD_PAS_MINI=0 , [DEFAULT]
pasmin , [R]

# If EVENEMENT = "INSTABILITY"
◇ACTION=/ "CONTINUE" , [DEFAULT]
/ "ARRET" ,
```

3) key words for the adaptation (if METHODE = "AUTO")

```
◇ ADAPTATION =_F (
◇EVENEMENT=/ "SEUIL" , [DEFAULT]
/"TOUT_INST"
/"NO"

# if EVENEMENT = "SEUIL"
◇ NB_INCR_SEUIL =/2 , [DEFAULT]
/NBINCSEUIL , [I]
◇ NOM_PARA = ' NB_ITER_NEWTON' [DEFAULT]
◇ CRIT_COMP = "IT" , [DEFAULT]
= "LT" ,
= "GE" ,
= "GT" ,
◇VALE_I= nbvalseuil , [I]
◇MODE_CALCUL_TPLUS = "FIXES" , [DEFAULT]
/"DELTA_Grandeur"
/"ITER_NEWTON"
/"IMPLEX"

# if MODE_CALCUL_TPLUS = "FIXES"
◇ PCENT_AUGM =/100 . , [DEFAULT]
/ pcent , [R]

# if MODE_CALCUL_TPLUS = "DELTA_Grandeur"
◆ VALE_REF =VALREF , [R]
◆ NOM_CHAM = "DEPL" ,
/"SIEF_ELGA" ,
/"VARI_ELGA" ,
◆ NOM_CMP =CMP , [TXT]

# if MODE_CALCUL_TPLUS = "ITER_NEWTON"
◆ NB_ITER_NEWTON_REF= nitref , [I]
```

```

)
◇INFO=/1 , , [DEFAULT]
/2 , [I]
)

```

3 Operands

3.1 Key word `DEFI_LIST`

3.1.1 “ manual

```

" OPERAND METHODE ◇METHODE=/, [DEFAULT]
/"CAR", [I]

```

THIS operand makes it possible to choose the mode of management of the list of times.

By default, management is manual (`METHODE=' MANUEL '`), which means that one will browse the list of times exactly defined by the user (see §3.1.23.1.2). In the event of under-cutting, new times can be inserted.

In automatic management (`METHODE=' AUTO '`), the code automatically calculates new times of computation. The modes of computation of new time step are defined under key word `ADAPTATION`. Times of computations defined by the user (see §3.1.23.1.2) are however respected.

3.1.2 Operands `LIST_INST` and `VALES`

```

◇/LISTE_INST =LIST , [l_r8]
/VALE =VALE , [R]
/◇RESULTAT =RESU , [result]
◇SUBD_PAS=NPAS [I]

```

times of computation can be defined in 3 different ways:

- either by informing key word `LIST_INST` with a concept defined as a preliminary by the operator `DEFI_LIST_REEL` [U4.34.01],
- or by informing the key word `VALE` with a list of realities,
- or by informing key word `RESULTAT` with a product concept by `STAT_NON_LINE` or `DYNA_NON_LINE`. This key word makes it possible to refine an already existing list of times. Indeed, one defines then a list of times `npas` **time finer** than the list of the times really calculated in the `STAT_NON_LINE` or `DYNA_NON_LINE`. It should be noted that this list of really calculated times can be different from the list of times given as starter from the `STAT_NON_LINE` or `DYNA_NON_LINE` (for example because of under-cutting) and from the list of filed times (if one does not file all times). This functionality is useful to check the independence of the results to the temporal discretization. For example, to obtain a list of times 2 times finer, one choose `SUBD_PAS=2`.

Caution! If the SD result `resu` comes from a computation with recovery, it is not possible to re-use it in the command because one cannot extract from list of single times (certain times will be in double).

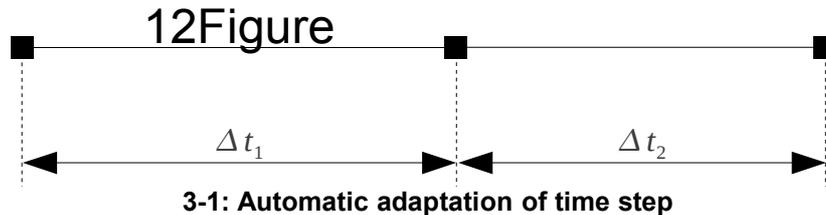
3.1.3 Operands `PAS_MINI`, `PAS_MAXI`, `NB_PAS_MAXI`

```

◇ PAS_MINI =PASMIN , [R]
◇ PAS_MAXI =PASMAY , [R]
◇NB_PAS_MAXI=NBPMASMAX , [I]

```

These operands are with being informed only in the event of automatic management of the list of times (METHODE = ' AUTO').



operand `PAS_MINI` makes it possible to specify the time step minimal one. So during the adaptation of time step, one is brought to treat one time step lower than `pasmin`, then computation stops. If this operand is not indicated, in general case `PAS_MINI` is worth then 10^{-12} (it is the authorized minimal value besides); if it is not indicated and that `MODE_CALCUL_TPLUS=' IMPLEX'`, then THE `PAS_MINI` is worth 1/1000ème first computation step (inevitably entered by the user *via* `LIST_INST`).

If $\Delta t_2 < \text{PAS_MINI}$: $\rightarrow \text{STOP}$

Attention, the checking of `PAS_MINI` is not carried out during the possible under-cutting of time step in the event of failure (it is `SUBD_PAS_MINI` which acts during under-cutting). Operand `PAS_MINI` acts **only** for the computation again time step.

Operand `PAS_MAXI` makes it possible to specify the time step maximum one, but its operation is different from that of `PAS_MINI`. Indeed, so during the adaptation of time step, one is brought to treat one time step higher than `pasmax`, then computation does not stop (contrary to the operation of `PAS_MINI`), but time step is then restricted with `pasmax`. If this operand is not indicated, this stage will not be made; if it is not indicated and that `MODE_CALCUL_TPLUS=' IMPLEX'`, then THE `PAS_MAXI` is worth 10 times the first computation step (inevitably entered by the user *via* `LIST_INST`).

If $\Delta t_2 > \text{PAS_MAXI}$: $\rightarrow \Delta t_2 = \text{PAS_MAXI}$

operand `NB_PAS_MAXI` allows to specify the number of time step beyond of which computation stops. The value by default is a million (it is also the authorized maximum value).

3.2 Key word ECHEC

This keyword factor répétable makes it possible to manage the list of times in the event of failure of convergence. For each occurrence key word `ECHEC`, one defines a cause of failure (see § 3.2.1) and an action to be carried out if this cause is satisfied (see § 3.2.2). The actions can be different between each occurrence from key word `ECHEC`.

If several causes of failure are satisfied simultaneously, the tripping order of these failures is predetermined:

"ERREUR" is priority on "DELTA_Grandeur"

3.2.1 Definition of the causes of failure

```

◇ EVENEMENT = "ERREUR", [DEFAULT]
              /"DIVE_RESI"
              /"DELTA_Grandeur"
              /"COLLISION"
              /"INTERPENETRATION"
              /"INSTABILITY"
    
```

operand `EVENEMENT` makes it possible to define the causes of failure.

3.2.1.1 Event "ERREUR"

Warning : The translation process used on this website is a "Machine Translation". It may be imprecise and inaccurate in whole or in part and is provided as a convenience.

By default, the cause of failure is `ERREUR`. This cause gathers the following reasons:

- Going beyond the authorized maximum number of iterations of Newton.
- Failure during the local integration of the constitutive law. The local parameters of the algorithm used for the integration of the constitutive law are in [U4.51.11].
- Failure of the algorithm of Deborst for the plane stresses or the models 1D. One will find more details in [U4.51.11].
- Failure during the resolution of the equation of control (see [U4.51.03]).
- Failure during the resolution of the problem of discrete contact (see [R5.03.50]). There is failure when the maximum number of iterations of contact is exceeded or when the contact matrix is singular.
- The matrix of the system is singular. The detection of singularity leans on parameter `NPREC` of the keyword factor `solver` (see [U4.50.01]).
- The iterative linear solver reached the maximum number of iterations of Krylov fixed by `NMAX_ITER` in the keyword factor `solver` (see [U4.50.01]).
- With convergence, physical criteria are not satisfied during integration with the constitutive law (notion of “vent-driven”)

Even if one finds no occurrence of `ECHEC/EVENEMENT = "ERREUR"`, the code automatically adds of them one with the current parameters by default (i.e. action “`DECOUPE`” in mode “`MANUEL`” and with `SUBD_NIVEAU=4`).

3.2.1.2 Event “`DIVE_RESI`”

the choice of event “`DIVE_RESI`” makes it possible to specify another possible cause of failure (“vent-driven”). There will be failure if the residue does not decrease sufficiently during iterations of Newton. More precisely, there will be failure if: $\min[R(i), R(i-1)] > R(i-2)$ for $i \geq 3$, where $R(j)$ is residue (`RESI_GLOB_MAXI`) with the iteration of Newton j . If there is failure, then time step under- is cut out. The interest this functionality is that it is not necessary to wait until one arrives at the end of the iterations of Newton authorized Re-to cut out time step in the event of “notable” NON-convergence. One realizes of NON-convergence more quickly, a time-saver is thus hoped.

Note:

The activation of this functionality with the discrete contact is not advised because in this kind of computation the residue does not decrease in general in a monotonous way.

3.2.1.3 Event “`DELTA_Grandeur`”

the choice of event “`DELTA_Grandeur`” makes it possible to specify another possible cause of failure (“vent-driven”). There will be failure if the increment of a component of a field exceeds a fixed threshold, **with convergence**. It is then necessary to define the name of the field by the operand `NOM_CHAM` (field of displacement, stresses or local variables), the name of the component by operand `NOM_CMP` and the value of the threshold of referenced by operand `VALE_REF`.

```
# if EVENEMENT = "DELTA_Grandeur"
  ◆VALE_REF=VALREF , [R]
  ◆ NOM_CHAM = "DEPL" ,
                /"SIEF_ELGA",
                /"VARI_ELGA",
  ◆NOM_CMP=CMP , [TXT]
```

more precisely, are v component `NOM_CMP` of field `NOM_CHAM` and *valref* the value of reference `VALE_REF`.

If field `NOM_CHAM` is a field at nodes and N_{no} is the set of all the nodes of the mesh, then there will be failure so with convergence:

$$\max(\Delta|v^i|, \forall i \in N_{no}) > valref$$

If field `NOM_CHAM` is a field with Gauss points, $N_{sp,pg,el}$ is the set of all the subpoints of all Gauss points of all the elements of the model, then there will be failure so with convergence:

$$\max(\Delta|v|, \forall i \in N_{sp,pg,el}) > valref$$

The increment Δ is to be taken with the meaning increment between two times.

N°1 example with a model `VMIS_ISOT_LINE` :

```
ECHEC=_F (
            EVENEMENT=' DELTA_Grandeur',
            VALE_REF=0.1e-2
            ,
            NOM_CHAM=' VARI_ELGA',
            NOM_CMP=' V1'),
```

One will have failure if the increment of the plasticity cumulated between two time step exceeds 0,1%.

N°2 example:

```
ECHEC= (
    _F (
            EVENEMENT=' DELTA_Grandeur',
            VALE_REF=5.e-2
            ,
            NOM_CHAM=' DEPL',
            NOM_CMP=' DX',),
    _F (
            EVENEMENT=' DELTA_Grandeur',
            VALE_REF=5.e-2
            ,
            NOM_CHAM=' DEPL',
            NOM_CMP=' DY'),
)
```

One will have failure if the norm (norm sup) of displacement (in 2D) varies of more than 5th-2 between two time step.

Note: in the two preceding examples, although `EVENEMENT=' ERREUR'` is not present, it will nevertheless be activated into tough.

3.2.1.4 Event "COLLISION"

This event makes it possible to cut out time step if the contact is detected (impact). It is a particularly useful functionality in dynamics. Indeed, in this case, an unsuited temporal discretization will reveal inopportune oscillations of displacement and velocity at the time of the shock. To minimize these numerical artefacts, one in general recommends to cut out finely time step at the time of the collision, which is possible only if for the typical cases where this time is detectable *a priori* by the user. In the general case, one can use this functionality so that cutting is done automatically.

3.2.1.5 Event "INTERPENETRATION"

◆ `PENE_MAXI` = `penmax` , [R]

This event detects the interpenetration of a node in a surface when the unilateral contact of regularized type (penalization) is activated. The user gives a value of the maximum penetration which he wants to obtain (parameter `PENE_MAXI`) and the code adapts the coefficient of penalization then (`ACTION=' ADAPT_COEF_PENA'`) or stops (`ACTION=' ARRET'`) to respect this stress. This functionality is available only for the contact in `FORMULATION=' DISCRETE'` (see `DEFI_CONTACT [U4.44.11]`).

3.2.1.6 Event "INSTABILITY"

This event makes it possible to manage the course of a nonlinear computation when an instability is detected at the time of the call to CRIT_STAB (cf [U4.51.03] and [U4.53.01]). If this key word CRIT_STAB is not used then this event cannot occur.

If an instability is detected, an action will then be started, according to two possibilities:

- ACTION=' CONTINUE ' : it is the value by default and one will continue computation, even in the event of instability,
- ACTION=' ARRET ' : if an instability is collected, then one will adopt the nonlinear resolution properly (the base will be saved and reusable in poursuite).

More details on the criterion of instability and its use are given in documentations [R7.05.01] and [U2.08.04].

3.2.2 Definition of the actions to be carried out in the event of failure

```

◇ ACTION = "DECOUPE", [DEFAULT]
           /"ARRET",
           /"REAC_PRECOND",
           /"ITER_SUPPL",
           /"AUTRE_PILOTAGE",
           /"ADAPT_COEF_PENA"
           /"CONTINUE"
    
```

According to the event, only certain actions are possible, see table below:

ACTION →	ARRET	DECOUPE	REAC_PRECOND	ITER_SUPPL	AUTRE_PILOTAGE	ADAPT_COEF_PENA	CONTINUE
EVENEMENT ↓							
ERREUR	YES	YES	YES	YES	YES	NON	NON
DIVE_RESI	NON	YES	NON	NON	NON	NON	NON
DELTA_Grandeur	YES	YES	NON	NON	NON	NON	NON
COLLISION	YES	YES	NON	NON	NON	NON	NON
INTERPENETRATI ON	YES	NON	NON	NON	NON	YES	NON
INSTABILITY	YES	NON	NON	NON	NON	NON	YES

For example, in the most general case (EVENEMENT=' ERREUR '), there are five possibilities:

- To cut out time step (ACTION=' DECOUPE ');
- To reactualize the pre-conditioner of the iterative solver (ACTION=' REAC_PRECOND ');
- To authorize additional iterations of Newton (ACTION=' ITER_SUPPL ');
- To change the solution from control (ACTION=' AUTRE_PILOTAGE ');
- To stop (ACTION=' ARRET ').

3.2.2.1 Action "ARRET"

If one chooses ACTION = "ARRET", computation stops properly: the base is saved and will be then reusable in poursuite.

3.2.2.2 Action "DECOUPE"

```

◇ SUBD_METHODE=/ "MANUEL"
                 /"AUTO"
# If ACTION = "DECOUPE" AND IF SUBD_METHODE = "AUTO"
  ◆SUBD_INST= subd_inst, [R]
  ◆SUBD_DUREE= SUBD_DUREE, [R]

# If SUBD_METHODE = "MANUEL"
  ◇SUBD_PAS=4 , [DEFAULT]
              not, [R]
  ◇SUBD_NIVEAU=3 , [DEFAULT]
    
```

Warning : The translation process used on this website is a "Machine Translation". It may be imprecise and inaccurate in whole or in part and is provided as a convenience.

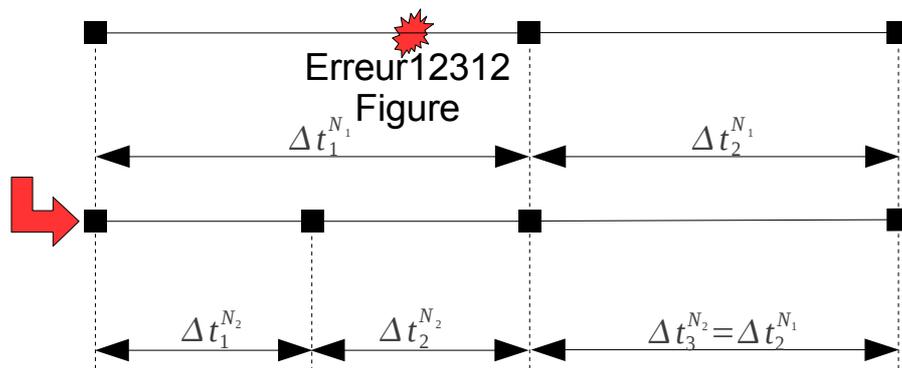
```

                                niv,                [I]
    ◇SUBD_PAS_MINI=0            .,                [DEFAULT]
                                pasmin,            [R]
# If SUBD_METHODE = "AUTO"
    ◇SUBD_PAS_MINI=0            .,                [DEFAULT]
                                pasmin,            [R]
    
```

This action consists in remaking time step by cutting out it. There exist two methods of cutting of time step: the mode `AUTO` and mode `MANUEL`.

Cut out in mode `MANUEL`

In mode `MANUEL`, time step is cut out in a uniform way. It is thus necessary to specify the number of cuttings using keyword `SUBD_PAS`.



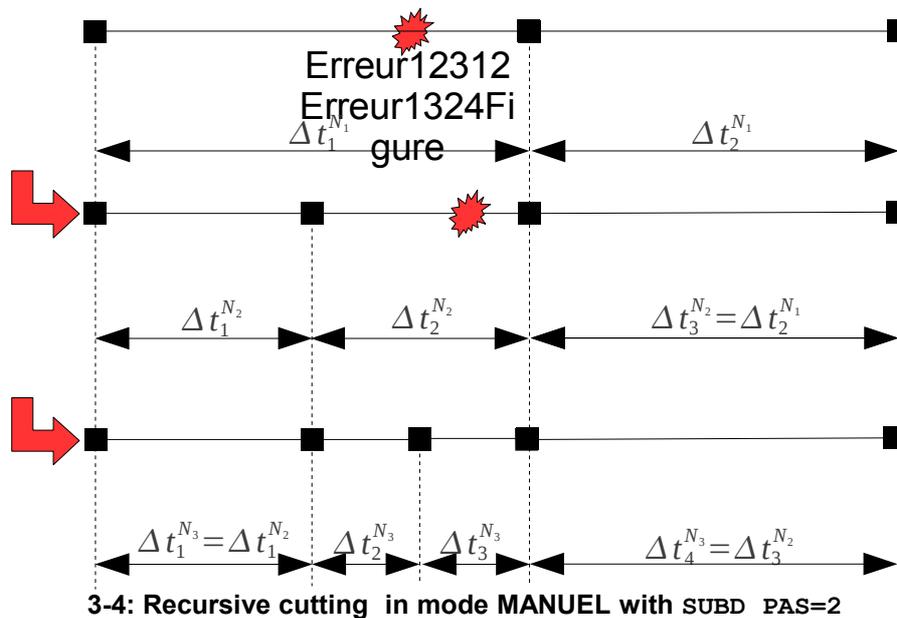
3-2: Cut out in mode `MANUEL` with `SUBD_PAS=2`

One sees an example on figure 3-2. Event `ERREUR` starts during time step the 1, initially length $\Delta t_1^{N_1}$. This step is then cut out in two steps (`SUBD_PAS = 2`) equal lengths such as $\Delta t_1^{N_2} = \Delta t_2^{N_2} = \frac{\Delta t_1^{N_1}}{2}$. The following step (number 2 in the initial configuration) becomes step 3 then, its length remains unchanged $\Delta t_3^{N_2} = \Delta t_2^{N_1}$.

Not	1	2	3
Initial - 1	$\Delta t_1^{N_1}$	$\Delta t_2^{N_1}$	
Cutting - 2	$\Delta t_1^{N_2}$	$\Delta t_2^{N_2}$	$\Delta t_3^{N_2}$

the notion of level of cutting

cutting of time step is recursive: if event `ERREUR` starts during a step already cut out, one redécoupe. One sees an example on figure 3-3.



After the first cutting, the failure occurs in (new) the second time step. One redécoupe this one with $\Delta t_2^{N_3} = \Delta t_3^{N_3} = \frac{\Delta t_2^{N_2}}{2}$. Time step according to unchanged rest $\Delta t_4^{N_3} = \Delta t_3^{N_2}$, the first time step is converged and thus remains also unchanged $\Delta t_1^{N_3} = \Delta t_1^{N_2}$. What gives:

Not	1	2	3	4
Initial – 1	$\Delta t_1^{N_1}$	$\Delta t_2^{N_1}$		
Cutting – 2	$\Delta t_1^{N_2}$	$\Delta t_2^{N_2}$	$\Delta t_3^{N_2}$	
Cutting – 3	$\Delta t_1^{N_3}$	$\Delta t_2^{N_3}$	$\Delta t_3^{N_3}$	$\Delta t_4^{N_3}$

There cannot be a variation of more than one level between two time step.

Cut out in mode AUTO

the automatic mode of cutting uses two possibilities:

- If the event release is a collision (EVENEMENT=' COLLISION'), then time step is cut out while having for purpose to have time step length `subd_inst`, maintained for the length of time `subd_duration`. If the time of cutting is too close to the collision ("shaving" contact too), cutting will not be effective, it is necessary then to change the initial discretization or the parameter `subd_inst` (if the contact becomes shaving after cutting);
- If not, one tries a linear extrapolation of the value of the residues.

If one notes *iter* the iteration of Newton and *resi^{ext}* the value of the extrapolated residue (RESI_GLOB_RELA and/or RESI_GLOB_MAXI), then one seeks *a*₁, *a*₂ and *b* such as:

$$resi^{ext} = \frac{(a_1 \cdot iter + a_2)}{b} \quad (1)$$

Here the algorithm used:

```
For iter = deb, fin
  xx = log(resi_iter+1)
  If iter > fin - 3
```

```
        poids=2
    If not
        poids=1
    One calculates
        xn= xn+ poids
        sx= sx+ poids.xx
        sy= sy+ poids.iter
        sxx= sxx+ poids.(xx2)
        syx= syx+ poids.xx.iter
Fine For
    a1= sxx.sy - sx.syx
    a2= -(sx.sy) + syx.xn
    b= -sx2 + sxx.xn
```

to make this extrapolation, one asks at least three iterations of Newton and one considers that two last iterations have twice more weight than first, if it is not possible (or if extrapolation fails for another reason), one passes automatically in mode of cutting `MANUEL (SUBD_PAS=4)` .

Control cutting

If convergence is difficult and that the event starts several times of continuation, involving the recursive cutting of time step, one can stop the process of two ways in mode `MANUEL` :

- With the keyword `SUBD_NIVEAU`, cutting will stop as soon as cutting is made more `niv` time.
- With the `SUBD_pas_mini` KEYWORD, cutting will stop as soon as time step is lower than `pasmin` .

In mode `AUTO`, only the keyword `SUBD_PAS_MINI` is available.

Notice concerning the computation of buckling:

During elastoplastic computation of buckling, it can arrive that the tangent matrix of the system is singular during iterations of Newton. While Re-cutting out time step, one can pass these hard points.

Characteristics of operand `SUBD_NIVEAU` :

- Operand `SUBD_NIVEAU` is a notion which should be total with all the occurrences of key word `ECHEC`. Thus, this operand should be defined only once. In the practice, it is not the case and nothing prevents from defining for each occurrence a value of `SUBD_NIVEAU` different from key word `ECHEC`. To secure this case there, the algorithm will retain with final like total and single value of `SUBD_NIVEAU` **the maximum** of the `SUBD_NIVEAU` defined by the user.

3.2.2.3 Action "REAC_PRECOND"

action "REAC_PRECOND" has meaning only if one uses an iterative linear solver ("PCG" or "PETSC") associated with the pre-conditioner single precision ("LDLT_SP").

In this case it allows the reactualization of the pre-conditioner after a failure of the linear resolution:
-with the first failure, one reactualizes the pre-conditioner then one remakes the same one time step (not cutting);
-if time step fails again, then one reactualizes the pre-conditioner and one cuts out

Note::

This action makes it possible to pass from computations which have strong specific non-linearities of time step to another but which, the rest of time, can be satisfied with a constant pre-conditioner.

It on the other hand is not adapted to computations presenting of strong non-linearities within same time step like computations of contact. For this kind of computation it is in general necessary to force the reactualization of the pre-conditioner by means of key word `REAC_PRECOND` of solver (cf [U4.50.01]).

3.2.2.4 Action "AUTRE_PILOTAGE"

action "AUTRE_PILOTAGE" makes it possible to define an action in a computation which calls on the control of the loading. During control, one can have to choose between two solutions (confer to the documentation [R5.03.80]) which one often discriminates using a residual criterion.

If the user selects action "AUTRE_PILOTAGE" and that convergence fails with the choice of solution which minimizes the residue, one starts again time step running, after it to have re-initialized correctly, but without subdividing. With the first iteration where the choice arises, one will choose the solution of maximum residue during this second attempt. One subdivides only if the two attempts failed.

This action will thus be undertaken only if the maximum number of iterations of Newton is reached and if it is the first attempt.

3.2.2.5 Action "ITER_SUPPL"

```
◇ PCENT_ITER_PLUS = 50 , [DEFAULT]
                    PCENT , [I]
◇ SUBD_METHODE = / "MANUEL" , [DEFAULT]
                  / "AUTO" ,
```

action "ITER_SUPPL" makes it possible to make iterations of Newton beyond the maximum given by `ITER_GLOB_MAXI` (or `ITER_GLOB_ELAS`). It prevents that computation fails for some iterations of Newton. For that, thanks to the linear extrapolation of the residues (see § 3.2.2.2), it is possible to consider the nombre of iterations of Newton necessary to convergence.

The nombre of iterations additional is restricted by the key word `PCENT_ITER_PLUS`. If the number estimated of iterations is higher than `pcent*ITER_GLOB_*`, then one considers that one failed. In the same way if extrapolation does not succeed. In these two cases, it is possible to activate the cutting of time step (keyword `SUBD_METHODE`).

3.2.2.6 Action "ADAPT_COEF_PENA"

```
◇ COEF_MAXI = 1E12 [DEFAULT]
             coemax, [R]
```

If event `INTERPENETRATION` starts, this action consists in automatically increasing the value of the coefficient of penalization (see operator `DEFI_CONTACT` [U4.44.11]) so as to decrease this interpenetration. Parameter `COEF_MAXI` makes it possible to specify the maximum value taken by the coefficient of penalization when the automatic adaptation mechanism is used, this in order to avoid problems of conditioning of the matrix (see the precautions of use of the methods penalized in [U2.04.04]).

Currently the mechanism does nothing but increase this coefficient and never makes it decrease, even if the interpenetration does not appear any more after adaptation, it thus tends to transform a problem of contact with interpenetration into problem without contact. It is thus advisable to choose a coefficient of penalization initial weaker than necessary (see operator `DEFI_CONTACT` [U4.44.11]).

3.2.2.7 Action "CONTINUE"

If the event `INSTABILITY` starts, this action consists in trying to continue nonlinear computation. It is the value by default for this event, which is the only one with being able to combine with this action.

3.3 Key word ADAPTATION

the adaptation of time step consists in calculating new time step Δt^{i+1} , starting from information of time step current Δt^i (and the time step preceding ones Δt^{i-1} Δt^{i-0} , ...).

Warning : The translation process used on this website is a "Machine Translation". It may be imprecise and inaccurate in whole or in part and is provided as a convenience.

The goal is thus to calculate $\Delta t^{i+1} = c \cdot \Delta t^i$, where c is a real coefficient.

The process is the following:

For the occurrence k of key word ADAPTATION, one defines an event and a mode of computation of Δt^{i+1} . If the event is checked, then one calculates c^k with the selected way of calculating.

A summary table of this type is obtained:

Occurrence n°1	$c^1 = 4,2$
Occurrence n°2	event not checked
Occurrence n°3	$c^3 = 1,7$
Occurrence n°4	$c^4 = 3,9$

Then, one chooses $c = \min(c^k)$ among the occurrences whose event is checked. In this example, the new one time step will be worth $\Delta t^{i+1} = 1,7 \cdot \Delta t^i$.

To use MODE_CALCUL_TPLUS=' IMPLEX', it is necessary that this key word is used only once.

3.3.1 Operand EVENEMENT

```

◇ EVENEMENT      =      "SEUIL",                               [DEFAULT]
                   /"TOUT_INST"
                   /"NO"
    
```

This operand makes it possible to specify the triggering function for the adaptation of time step.

If EVENEMENT=' SEUIL', then the event is checked only if the threshold is crossed.

If EVENEMENT=' TOUT_INST', then the event is checked at all times; this case is compulsory with method IMPLEX.

If EVENEMENT=' AUCUN', then the event is never checked.

3.3.2 Operands NB_INCR_SEUIL, NOM_PARA, CRIT_COMP, VALE_I

```

◇ NB_INCR_SEUIL  = /2 ,                                       [DEFAULT]
                   /NBINCSEUIL,                               [I]
◇ NOM_PARA       = ' NB_ITER_NEWTON'                          [DEFAULT]
◇ CRIT_COMP      = "IT",                                       [DEFAULT]
                   = "LT",
                   = "GE",
                   = "GT",
◇ VALE_I         = nbvalseuil,                                  [I]
    
```

These operands are with being informed only in the event of event of type threshold (EVENEMENT=' SEUIL '). The event is checked if there is nbincseuil time of continuation a "green light", a green light being characterized by condition NOM_PARA CRIT_COMP VALE_I.

For example if NB_INCR_SEUIL=2 , NOM_PARA=' NB_ITER_NEWTON' , CRIT_COMP='LE' and VALE_I=5 , then the event are checked if one made 2 times of continuation less than 5 iterations of Newton.

Note:

There is no default value for VALE_I in the catalog of operator DEFILIST_INST . However, if its value is not indicated, one will take for VALE_I half of the maximum number of iterations of Newton (declared in STAT_NON_LINE/CONVERGENCE/ITER_GLOB_MAXI).

3.3.3 Operand MODE_CALCUL_TPLUS

```

◇ MODE_CALCUL_TPLUS = "FIXES", [DEFAULT]
                    / "DELTA_GRANDEUR"
                    / "ITER_NEWTON"
                    / "IMPLEX"
    
```

This operand is used time step to specify the mode of computation of new (in fact, of the coefficient c).

If MODE_CALCUL_TPLUS='FIXE', then the coefficient c is fixed. Its value is given thanks to operand PCENT_AUGM. The new one time step will be worth then $\Delta t^{i+1} = (1 + \frac{\text{pcent_augm}}{100}) \Delta t^i$.

If MODE_CALCUL_TPLUS='DELTA_GRANDEUR', then the coefficient c is related to the variation during time step of a selected quantity. The following formula is used:

$c = \min_{i \in N} \left(\frac{V_{ref}}{|\Delta V_i|} \right)$ with N all the nodes or Gauss points for which ΔV_i is not null. The value of reference V_{ref} is chosen by the user.

If the field ΔV is identically null, then it is considered that the criterion (the event) is not checked.

If MODE_CALCUL_TPLUS='ITER_NEWTON', then the coefficient c is related to the variation during time step amongst iterations of Newton. The following formula is used:

$c = \sqrt{\frac{N_{ref}}{N_{it} + 1}}$ where N_{it} is the nombre of iterations of Newton carried out and N_{ref} a value of reference chosen by the user. It is pointed out that the prediction is regarded as the iteration of Newton n°0, which means that if one converges as of the prediction then $c = \sqrt{N_{ref}}$.

If MODE_CALCUL_TPLUS='IMPLEX', then the coefficient c is related to the increments of local variables (cumulated plastic strain or damage) of the preceding step (see [R5.03.81]). In this case, key word ADAPTATION should be used only one times. The coefficient c inevitably will lie between 0,5 and 1,2. Moreover, by definition of method IMPLEX there will be no cause of failure.

3.3.4 Operand PCENT_AUGM

```

◇ PCENT_AUGM = /100 ., [DEFAULT]
              /PCENT , [R]
    
```

This operand is used to specify the percentage of increase in time step if MODE_CALCUL_TPLUS = "FIXES". By default, one increases by 100%, i.e. one doubles the value of time step. The value can be negative and must be strictly higher than -100.

3.3.5 Operands VALE_REF, NOM_CHAM, NOM_CMP

```

◆ VALE_REF = VALREF , [R]
◆ NOM_CHAM = "DEPL",
              / "SIEF_ELGA",
              / "VARI_ELGA",
◆ NOM_CMP = CMP , [K8]
    
```

These operands are used to specify the computation of the coefficient c of increase in time step if `MODE_CALCUL_TPLUS = "DELTA_Grandeur"` (cf §3.3.33.3.3) with `VALE_REF` the `NOM_CHAM` and V_{ref} value of reference the name of the field of variables considered.

3.3.6 Operand `NB_ITER_NEWTON_REF`

◆ `NB_ITER_NEWTON_REF=NITREF` , [I]

This operand is used to inform the nombre of iterations of Newton as reference, noted N_{ref} (cf §3.3.33.3.3) if `MODE_CALCUL_TPLUS = "ITER_NEWTON"`.

3.4 Operand `INFO`

◇ `INFO`

For the moment, this operand is not useful.