Procédure TEST_COMPOR

1 But

Cette macro-commande permet de tester les lois de comportement mécaniques du point de vue de la robustesse et de la fiabilité :

- dans le cas mécanique, le test met en œuvre une simulation d'un trajet de chargement multi-axial sur un point matériel, dans le but de vérifier la robustesse de l'intégration numérique des comportements testés, de leur insensibilité par rapport à un changement d'unités, de l'invariance du résultat par rapport à une rotation globale appliquée au problème, la justesse de la matrice tangente ;
- dans le cas des variables de commandes (température, …) ce test vérifie la bonne prise en compte des variables de commande dont dépendent les coefficients du modèle, ainsi que les termes de déformation qui en résultent.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

2 Syntaxe

```
TEST_COMPOR  (
      ◊  OPTION    =  / 'THER',                                  [DEFAUT]
                      / 'MECA',
      ♦  COMPORTEMENT     =_F (voir le document [U4.51.11] ),

      ◊  NEWTON           =_F (  voir le document [U4.51.03]),

       ◊ CONVERGENCE      = _F(
         /RESI_GLOB_RELA     =  1.E-6,                          [DEFAUT]
         /|RESI_GLOB_MAXI    =  resmax,                         [R]
          | RESI_GLOB_RELA      =  resrel,                        [R]
          ITER_GLOB_MAXI     =  /10,                            [DEFAUT]
                                /maglob,                        [I]
                         ),
      si OPTION = 'THER'

         ♦  MATER         =  mater,                             [mater]
         ♦  LIST_MATER    =  mater,                             [l_mater]
         ◆  ALPHA         =  alpha,                             [fonction]
         ◆  YOUNG         =  young,                             [fonction]
         ◆  TEMP_INIT     =  temp_init,                         [R]
         ◆  TEMP_FIN      =  temp_fin,                          [R]
         ◆  NB_VARI       =  nb_vari,                           [I]
         ◊  SUPPORT        = /'POINT'                           [DEFAUT]
                            /'ELEMENT'
         ◊  INST_FIN      =  temp_fin,                          [R]
         ◊  VARI_TEST     =  vari_test,                         [Kn]
         ◊  D_SIGM_EPSI   =  d_sigm_epsi,                       [fonction]
         ◊  C_PRAG        =  c_prag,                            [fonction]

      si OPTION = 'MECA'
         ♦  LIST_MATER    =  mater,
         ◆  POISSON       =  poisson,                              [R]
         ◆  YOUNG         =  young,                                [R]
         ◊  LIST_NPAS     =  list_npas,                            [l_I]
         ◊  LIST_TOLE     =  list_tole,                            [l_R]
         ◊  PREC_ZERO     =  prec_zero,                            [l_R]
         ◊  VARI_TEST     =  vari_test,                            [Kn]
         ◊  SUPPORT        =  /'POINT'                             [DEFAUT]
                             /'ELEMENT'
         ◊  MODELISATION  =  /'3D'                                 [DEFAUT]
                             /'C_PLAN'
         ◊  MASSIF            = /'ANGL_REP'                       [R]
                              /'ANGL_EULER'                       [R]
         ◊  ANGLE          = angz,                                [R]
         ◊  VERI_MATR_OPTION =_F(
             ◊ VALE_PERT_RELA  =                                  [R]
             ◊ PRECISION       =                                  [R]
             ◊ PREC_ZERO       =                                  [R]
         ),

      ◊  INFO =     /  1 ,                                        [DEFAUT]
                    /  2 ,

            )
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

3 Opérandes

3.1 Mots-clés COMPORTEMENT/NEWTON

La syntaxe de ces mots clés est décrite dans le document [U4.51.03] et [U4.51.11].

3.2 Mot clé CONVERGENCE

◊ CONVERGENCE =_F()

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

3.2.1 Opérande RESI_GLOB_RELA/RESI_GLOB_MAXI

◊ |RESI_GLOB_RELA = resrel , [R]

L'algorithme continue les itérations globales tant que :

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

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

Lorsque le chargement et les réactions d'appui deviennent nuls, c'est-à-dire lorsque $\boldsymbol{L}$ est nul (par exemple dans le cas d'une décharge totale), on essaie de passer du critère de convergence relatif RESI_GLOB_RELA au critère de convergence absolu RESI_GLOB_MAXI. Cette opération est transparente pour l'utilisateur (message d'alarme émis dans le fichier .mess). Lorsque le vecteur $\boldsymbol{L}$ redevient différent de zéro, on repasse automatiquement au critère de convergence relatif RESI_GLOB_RELA.
Toutefois, ce mécanisme de basculement ne peut pas fonctionner au premier pas de temps. En effet, pour trouver une valeur de RESI_GLOB_MAXI raisonnable de manière automatique (puisque l'utilisateur ne l'a pas renseigné), on a besoin d'avoir eu au moins un pas convergé sur un mode RESI_GLOB_RELA . Dès lors, si le chargement est nul dès le premier instant, le calcul s'arrête. L'utilisateur doit déjà alors vérifier que le chargement nul est normal du point de vue de la modélisation qu'il réalise, et si tel est le cas, trouver un autre critère de convergence (RESI_GLOB_MAXI par exemple).

Si cet opérande est absent, le test est effectué avec la valeur par défaut, sauf si RESI_GLOB_MAXI est présent.

◊ |RESI_GLOB_MAXI = resmax , [R]

L'algorithme continue les itérations globales tant que :

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

où $\boldsymbol{F}^n$ est le résidu de l'itération n (Cf. [R5.03.01] pour plus de détails). Si cet opérande est absent, le test n'est pas effectué.

Si RESI_GLOB_RELA et RESI_GLOB_MAXI sont présents tous les deux, les deux tests sont effectués.

3.2.2 Opérande ITER_GLOB_MAXI

◊ ITER_GLOB_MAXI = /10 [DEFAUT]
/maglob

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

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

3.3 OPTION='THER'

Test thermo-mécanique permettant de valider la prise en compte de la variation de température dans les lois de comportement (cf. *V6.07.108)*. Ces tests permettent de vérifier les deux points suivants :

- La dilatation thermique est bien calculée (avec prise en compte de la variation de la dilatation thermique avec la température)
- La variation des coefficients matériau avec la température est correcte, en particulier dans la résolution incrémentale du comportement.

Il s'agit d'une double simulation, la première en thermomécanique, la seconde en mécanique pure. La première sera validée en comparaison de la seconde, en supposant bien sûr que le comportement testé fournit une solution correcte en mécanique pure.

La première simulation (solution que l'on cherche à valider) consiste à appliquer une variation de température sur un point matériel, en bloquant par exemple les déformations suivant x : $\varepsilon_{xx}=0$. La température imposée est croissante linéairement en fonction du temps.

La seconde simulation (qui doit être équivalente à la première) consiste à appliquer au même point matériel une déformation imposée suivant x : $\varepsilon_{xx}=-\varepsilon^{th}=-\alpha(T)(T-T_{ref})$, en mécanique pure. En effet, pour tout comportement (en supposant la décomposition additive des déformations) :

$$\sigma_{xx}=E(T)(\varepsilon_{xx}-\varepsilon^{th}-\varepsilon^{p}_{xx})$$

dans le premier cas, $\sigma_{xx}=E(T)(0-\varepsilon^{th}-\varepsilon^{p}_{xx})$, et dans le second : $\sigma_{xx}=E(T)(\varepsilon-\varepsilon^{p}_{xx})$.

Il suffit donc, à chaque instant d'appliquer, pour le calcul mécanique, $\varepsilon_{xx}=-\varepsilon^{th}=-\alpha(T)(T-T_{ref})$. De plus, pour obtenir les mêmes résultats dans les deux cas, il est nécessaire, à chaque pas de temps de la seconde simulation, d'effectuer le calcul mécanique pur avec des coefficients dont les valeurs sont interpolées en fonction de la température à l'instant courant (opérande list_mater).

3.3.1 Opérande MATER

♦ MATER = mater,

Ce mot-clé permet de renseigner le nom du matériau (mater) défini par DEFI_MATERIAU [U4.43.01], où sont fournis les paramètres nécessaires au comportement choisi, fonctions de la température.

3.3.2 Opérande LIST_MATER

♦ **LIST**_MATER =list_mater,

Ce mot-clé permet de renseigner une liste de matériaux (list_mater), définis par DEFI_MATERIAU [U4.43.01], dont les paramètres constants correspondent à ceux de mater, interpolés en fonction de la température.

3.3.3 Opérandes ALPHA / YOUNG

◆ ALPHA = alpha, [fonction]
◆ YOUNG = young, [fonction]

Ces mot-clés permettent de renseigner le module d'Young et le coefficient de dilatation thermique fonctions de la température, afin de calculer les déformations thermiques et les contraintes correspondantes.

3.3.4 Opérandes TEMP_INIT / TEMP_FIN / INST_FIN

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

◆ TEMP_INIT = temp_init, [R]
◆ TEMP_FIN = temp_fin, [R]
◊ INST_FIN = temp_fin, [R]

Ces mot-clés permettent de renseigner les températures initiale et finale, ainsi que l'intant final du transitoire (correspondant à temp_fin), valant 1. par défaut.

3.3.5 Opérandes NB_VARI / VARI_TEST

◆ NB_VARI = nb_vari, [I]
◊ VARI_TEST = vari_test, [Kn]

Ces mot-clés permettent de renseigner le nombre de variables internes du comportement choisi, ainsi que les variables internes à tester (par défaut, toutes les variables internes sont testées).

3.3.6 Opérandes D_SIGM_EPSI / C_PRAG

◊ D_SIGM_EPSI = d_sigm_epsi, [fonction]
◊ C_PRAG = c_prag, [fonction]

Dans le cas particulier des comportements à écrouissage cinématique linéaire, ces mots clés permettent de définir la pente d'écrouissage cinématique en fonction de la température. Cette pente vaut :

- d_sigm_epsi pour le comportement VMIS_CINE_LINE,
- c_prag pour les comportements VMIS_ECMI_LINE, VMIS_ECMI_TRAC.

3.3.7 Opérande SUPPORT

◊ SUPPORT = /'POINT' [DEFAUT]
/'ELEMENT'

Voir [U4.51.12]

3.4 OPTION = 'MECA'

Test mécanique pur, qui met en œuvre une simulation d'un trajet de chargement en déformations en un point matériel, c'est à dire sur un modèle tel que les états de contraintes et de déformations sont homogènes à tout instant. Il permet ainsi de tester un certain nombre de modèles de comportement, dans le but de vérifier la robustesse de leur intégration numérique, leur insensibilité par rapport à un changement d'unités, l'invariance par rapport à une rotation globale appliquée au problème, la justesse de la matrice tangente. Ce test procède, pour chaque modélisation, à une inter-comparaison entre la solution de référence (obtenue avec un pas de temps très fin), la solution avec une discrétisation moyennement grossière, la solution avec effet de la température (ou d'une autre variable de commande), la solution en changeant le système d'unités (Pa en MPa), et celle obtenue après rotation ou symétrie (voir le document [v6.07.101]).

3.4.1 Opérande LIST_MATER

♦ LIST_MATER =list_mater,

Ce mot-clé permet de renseigner une liste de 2 matériaux (list_mater), définis par DEFI_MATERIAU [U4.43.01], dont les paramètres constants sont évalués soit en Pa , soit en Mpa .

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

3.4.2 Opérandes POISSON / YOUNG

◆ POISSON = poisson, [R]
◆ YOUNG = young, [R]

Ces mot-clés permettent de renseigner le module d'Young et le coefficient de Poisson.

3.4.3 Opérandes LIST_NPAS / LIST_TOLE

◊ LIST_NPAS = list_npas, [l_I]
◊ LIST_TOLE = list_tole, [l_R]

Ces mot-clés permettent de renseigner la discrétisation en temps et les tolérances correspondantes.

Par défaut, $\text{list_npas}=[1,1,1,1,1,5,25]$ (4 problèmes “équivalents” avec la discretisation la plus grossière, soit 1 incrément par segment de chargement, puis variation de la discrétisation : 1 puis 5 puis 25 incréments par segment).

Par défaut, $\text{list_tole}=4\times[1.\text{E}-10]+[1.\text{E}-1]+2*[1.\text{E}-2]+[1.\text{E}-8]$. La précision requise pour les problèmes équivalent est volontairement très petite (sinon il y a un risque de bug). Les précisions suivantes sont plus lâches, puisque les comporements sont en général sensibles à la discrétisation en temps. La denière valeur est la tolérance sur la matrice tangente.

3.4.4 Opérande PREC_ZERO

◊ PREC_ZERO = prec_zero, [l_R]

Ce mot-clé permettent de fournir un zéro “numérique” pour chaque variable testée, afin de calculer une erreur relative significative. prec_zero a donc la même longueur que vari_test. Par défaut cette liste vaut : $3\times1.\text{E}-10$.

3.4.5 Opérande VARI_TEST

◊ VARI_TEST = vari_test, [Kn]

Liste des composantes testées, supposées invariantes dans les problème équivalents (rotation, changement d'unité). Par défaut vari_test = ('V1',VMIS','TRACE').

3.4.6 Opérande SUPPORT

◊ SUPPORT = /‘POINT’ [DEFAUT]
/'ELEMENT'

Voir [U4.51.12]

3.4.7 Mot-clé MODELISATION

Le mot-clé MODELISATION permet, dans le cas SUPPORT='ELEMENT', d'effectuer le calcul sur un élément 3D ou sur un élément 2D, en contraintes planes. Il n'est pas disponible dans le cas SUPPORT='POINT', car il suffit d'imposer une valeur nulle aux composantes correspondantes aux contraintes planes ou aux déformations planes pour obtenir le même résultat.

3.4.8 Mot-clé ANGLE

Ce mot-clé permet de spécifier un angle (en degrés) pour effectuer une rotation d'ensemble autour de Z appliquée à la fois au chargement, au maillage, et au dépouillement. Ceci permet surtout de vérifier la fiabilité de l'intégration du comportement, comme dans les tests **COMP001, COMP002**.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Par défaut, la rotation est identiquement nulle.

Dans le cas de matériaux possédant une orientation intrinsèque (orthotropie, comportements cristallins), il convient d'utiliser également le mot-clé MASSIF, avec une première valeur d'angle identique à celle fournie sous ANGLE.

3.4.9 Mot-clés MASSIF / ANGL_EULER / ANGL_REP

Ces mot-clés permettent de définir une orientation intrinsèque au matériau (orthotropie, comportements cristallins), et permettent de faire appel dans la macro-commande au mot-clé **MASSIF** de **AFFE_CARA_ELEM** [U4.42.01] .
Par défaut, l'orientation est nulle, et on ne fait pas appel à **AFFE_CARA_ELEM**.

3.4.10 Mot-clés VERI_MATR_OPTION

Ce mot-clés sert à regrouper les opérandes qui gérent le comportement de TEST_COMPOR lors de la vérification de la matrice tangente analytique avec celle obtenue par perturbation numérique.

3.4.11 Opérande VALE_PERT_RELA

◊ VALE_PERT_RELA = [R]

Permet de définir la valeur de la perturbation relative numérique qui intervient dans le calcul de la matrice perturbée. Pour plus de détail se référer à [U4.51.11].

3.4.12 Opérande PRECISION

◊ PRECISION = [R]

L'opérande PRECISION permet de définir la valeur au dessus de laquelle on considére que la matrice analytique et la matrice perturbée sont différentes.

3.4.13 Opérande PREC_ZERO

◊ PREC_ZERO = [R]

En dessous de PREC_ZERO, on ne compare pas les valeurs des termes de la matrice tangente. Cela permet de gérer les situations où les termes de la matrice tangente perturbée sont très proches de zéro.

3.5 Opérande INFO

Précise le détail des informations imprimées dans le fichier message.

En mode INFO=2, on imprime toutes les tables produites par SIMU_POINT_MAT.

4 Exemple

4.1 OPTION='MECA'

Voir les tests COMP001 [V6,07,101] , COMP002 [V6,07,102]

```
#unités en Pa
ACIER[0]=DEFI_MATERIAU(ELAS=_F(E=YOUNG_Pa,
                               NU=POISSON,
                               ALPHA=11.8e-6),
                       ECRO_LINE=_F(D_SIGM_EPSI=pente_Pa,
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
                                    SY=SY_Pa,),);
#unités en MPa
ACIER[1]=DEFI_MATERIAU(ELAS=_F(E=YOUNG,
                               NU=POISSON,
                               ALPHA=11.8e-6),
                     ECRO_LINE=_F(D_SIGM_EPSI=pente,
                                  SY=SY,),)
compor='VMIS_ISOT_LINE'
tabresu=TEST_COMPOR(OPTION='MECA',
              COMPORTEMENT=_F(RELATION=compor,),
              NEWTON=_F(REAC_ITER=1),
              LIST_MATER=ACIER,
              VARI_TEST=('V1','VMIS','TRACE'),
              YOUNG=YOUNG,POISSON=POISSON,
              )
```

4.2 OPTION='THER'

```
Voir les tests COMP008*
TREF = 0.
Tmax = 500.
YOUN=DEFI_FONCTION(NOM_PARA='TEMP',VALE=(TREF, 200000.,
                                         Tmax, 100000.,
                                           ),);
ALPH=DEFI_FONCTION(NOM_PARA='TEMP',VALE=(TREF, 1.E-5,
                                       Tmax, 2.E-5,
                                           ),);
SIGY=DEFI_FONCTION(NOM_PARA='TEMP',VALE=(TREF, 100.,
                                         Tmax, 50.,
                                           ),);
DSDE=DEFI_FONCTION(NOM_PARA='TEMP',VALE=(TREF, 10000.,
                                         Tmax, 5000.,
                                           ),);
MATERI=DEFI_MATERIAU(ELAS_FO=_F(E=YOUN, NU=ZERO,
                                TEMP_DEF_ALPHA=TREF,
                                ALPHA=ALPH,),
                     ECRO_LINE_FO=_F(D_SIGM_EPSI=DSDE,
                                                SY=SIGY,),
                   );
LMAT2  = [None]*(NCAL)
time=0.
for i in range(NCAL):
    timem = time
    time = timem + tfin/NCAL
    Ti = T0 + time /tfin * (Tmax - T0)
    LMAT2[i]=DEFI_MATERIAU(ELAS =_F(E=YOUN(Ti),
                                       NU=0.,
                                       ALPHA=0.,),
                           ECRO_LINE=_F(D_SIGM_EPSI=DSDE(Ti),
                                                  SY=SIGY(Ti),),)
compor='VMIS_ISOT_LINE'

tabresu=TEST_COMPOR(
        MATER=MATERI,COMPORTEMENT=_F(RELATION=compor),
        LIST_MATER=LMAT2,ALPHA=ALPH,YOUNG=YOUN,TEMP_INIT=TREF,TEMP_FIN=Tmax,
        NEWTON=_F(REAC_ITER=1),
        NB_VARI=2,  VARI_TEST=('V1','V2'),
              )
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)