Opérateur CALC_G

1 But

Cet opérateur calcule les grandeurs de mécanique de la rupture suivantes, en 2D et en 3D :

- le taux de restitution d'énergie par la méthode thêta dans le cas d'un problème thermo-élastique linéaire ou non linéaire [R7.02.01] et [R7.02.03], en statique ou en dynamique [R7.02.02], et en élastoplasticité [R7.02.07] ;
- les facteurs d'intensité de contraintes $K1$, $K2$ et $K3$ par la méthode des déplacements singuliers dans le cas d'un problème thermo-élastique linéaire [R7.02.05] ;
- la maximisation de G et de $K1$ sous des contraintes bornes.

Cet opérateur peut être utilisé aussi bien pour des fissures maillées (approche classique) que pour des fissures non maillées (méthode X-FEM).

Avant une première utilisation, il est conseillé de se référer aux documents de référence et de conseils d'utilisation correspondants, notamment le document [U2.05.01].

Cet opérateur génère un concept de type `table_sdaster`.

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

2 Syntaxe

```
[table_sdaster] =  CALC_G

    (

#   Récupération du résultat du calcul mécanique
      ♦  RESULTAT  =  resu,                          /  [evol_elas]
                                                     /  [evol_noli]
                                                     /  [dyna_trans]
                                                     /  [mode_meca]

         #  Si RESULTAT de type evol_elas, evol_noli ou dyna_trans
         ◊  /  TOUT_ORDRE   =  'OUI',                [DEFAUT]
            /  NUME_ORDRE   =  l_ordre,              [l_I]
            /  LIST_ORDRE   =  lis,                  [listis]
            /  INST         =  l_inst,               [l_R]
            /  LIST_INST    =  l_reel,               [listr8]

         #  Si RESULTAT de type mode_meca
         ◊  /  TOUT_MODE    =  'OUI',                [DEFAUT]
            /  NUME_MODE    =  l_ordre,              [l_I]
            /  LIST_MODE    =  lis,                  [listis]
            /  FREQ         =  l_inst,               [l_R]
            /  LIST_FREQ    =  l_reel,               [listr8]

         ◊   CRITERE =   / 'RELATIF',                [DEFAUT]
                         ◊   PRECISION = / prec ,    [R]
                                         / 1.E-6 ,   [DEFAUT]
                         / 'ABSOLU' ,
                          ♦  PRECISION = prec ,      [R]


#  Récupération ou création du champ théta

    ♦ THETA  =_F(    ♦  /  ♦ FOND_FISS = ff,         [fond_fiss]
                        ◊  /  R_INF = r,             [R]
                           /  R_INF_FO = rz,         [fonction]
                        ◊  /  R_SUP = r,             [R]
                           /  R_SUP_FO = rz,         [fonction]
                            ◊ NB_POINT_FOND =  n,    [I]

                      /  ♦ FISSURE = ffx,            [fiss_xfem]
                        ◊  /  R_INF = r,             [R]
                           /  R_INF_FO = rz,         [fonction]
                        ◊  /  R_SUP = r,             [R]
                           /  R_SUP_FO = rz,         [fonction]

                        ◊  NUME_FOND =  / 1,         [DEFAUT]
                                        / n,         [I]
                        ◊  NB_POINT_FOND =  n,       [I]

               ),

#  Chargement
               ◊  EXCIT = _F ( ♦ CHARGE  =  charge ,  [char_meca]
                                                      [char_cine_meca]
```


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

```
                                ◊ FONC_MULT   =   fmult,     [fonction]
                                                             [formule]
                                ),
            #  Finsi
```

Comportement

```
        ◊   /   COMPORTEMENT =_F (
               ♦   RELATION      =   /   'ELAS',
                                     /   'ELAS_VMIS_LINE',
                                     /   'ELAS_VMIS_TRAC',
                                     /   'ELAS_VMIS_PUIS',
               ◊   DEFORMATION   =   /   'PETIT',              [DEFAUT]
                                     /   'GROT_GDEP',
                                     /   'PETIT_REAC',
               ◊   /   TOUT      =   'OUI',                  [DEFAUT]
                   /   GROUP_MA  =   lgrma,               [l_gr_maille]
                         ),
# si relation='ELAS'
        ◊   ETAT_INIT     =_F (SIGM  = siefelga,          [cham_elem, cham_no]
                             ),
```

Méthode de discrétisation de thêta en fond de fissure (3D local)

```
        ◊   LISSAGE = _F(◊   LISSAGE_THETA=   /   'LEGENDRE'       [DEFAUT]
                                              /   'LAGRANGE'
                             ◊   LISSAGE_G    =   /   'LEGENDRE'       [DEFAUT]
                                                  /   'LAGRANGE'
                                                  /   'LAGRANGE_NO_NO'
                      # Si LISSAGE_THETA = 'LEGENDRE' ou LISSAGE_G = 'LEGENDRE':
                             ◊   DEGRE  =   /   5                  [DEFAUT]
                                            /   0,1,2,3,4,6,7,
                             ),
```

Option demandée

```
        ♦   OPTION =   /   'CALC_G'  ,
                       /   'CALC_K_G',
```

Si OPTION = ‘CALC_G’

```
         ◊ CALCUL_CONTRAINTE   =   /   ‘ OUI ’                  [DEFAUT]
                                   /   'NON',


        ◊   TITRE  =      titre,                        [l_Kn]
```

Impression d’informations

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


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

3 Opérandes

3.1 Opérande RESULTAT

/ RESULTAT = resu

Nom d'un concept résultat de type evol_elas, evol_noli, dyna_trans ou mode_meca. Cet opérande permet de récupérer le champ de déplacement (et de vitesse et d'accélération pour un calcul en dynamique).

Le modèle et le champ de matériau, nécessaires au calcul, sont également extrait de la structure de données résultat. Les options de calcul possibles pour chaque type de modélisation sont rappelées dans le tableau ci-dessous.

	Calcul de G	Calcul de K
D_PLAN / C_PLAN, D_PLAN_INCO_UPG, D_PLAN_INCO_UP *Fissure maillée*	CALC_G	CALC_K_G
D_PLAN / C_PLAN *Fissure non maillée*	CALC_G	CALC_K_G
AXIS, AXIS_INCO_UPG, AXIS_INCO_UP *Fissure maillée*	CALC_G	CALC_K_G
AXIS *Fissure non maillée*	CALC_G	CALC_K_G
3D, 3D_INCO_UPG, 3D_INCO_UP *Fissure maillée*	CALC_G	CALC_K_G
3D *Fissure non maillée*	CALC_G	CALC_K_G

Tableau 3.1 : Disponibilité, par modélisation, des options de calcul.

Remarques sur les propriétés matériaux :

Les caractéristiques du matériau, récupérées dans la structure de données resu, sont les suivantes :
- module d'Young E ,
- coefficient de Poisson NU,
- coefficient de dilatation thermique ALPHA (pour un problème thermo-mécanique),
- limite d'élasticité SY (pour un problème élastique non linéaire),
- pente de la courbe de traction D_SIGM_EPSI (pour un problème élastique non linéaire avec écrouissage isotrope linéaire).

Pour le calcul de restitution d'énergie, ces caractéristiques peuvent dépendre de la géométrie (option 'CALC_G') et de la température (option 'CALC_G'). Elles doivent être indépendantes de la température pour le calcul des facteurs d'intensité de contraintes.

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

Les caractéristiques SY et D_SIGM_EPSI ne sont traitées que pour un problème élastique non linéaire avec écrouissage de Von Mises. Le calcul des coefficients d'intensité de contraintes est traité uniquement en élasticité linéaire.

Remarque :

Pour le calcul des facteurs d'intensité de contraintes (option 'CALC_K_G'), les caractéristiques doivent être définies sur tous les matériaux, y compris sur les éléments de bord, du fait de la méthode de calcul [R7.02.05]. Pour s'assurer de ce fait, il est conseillé de faire un AFFE = _F (TOUT = 'OUI') dans la commande AFFE_MATERIAU [U4.43.03], quitte à utiliser la règle de surcharge ensuite.

P our les éléments incompressibles (_INCO_UPG,_INCO_UP), il est conseillé d'utiliser STAT_NON_LINE pour obtenir les résultats.

Les facteurs d'intensité des contraintes obtenus avec l'option CALC_K_G sont calculés en évaluant la forme bilinéaire de G avec une solution singulière purement mécanique (solution asymptotique de Westergaard). Si on résout un problème thermo-mécanique, on ne prend alors pas en compte la singularité due au champ thermique.

Un indicateur de l'erreur due à cette approximation peut être obtenu en évaluant la différence entre G et G_IRWIN . En pratique, on évalue en tout point du fond de fissure la quantité $\frac{|G-G_{irwin}|}{|G|}$ *, et on en fait ensuite la moyenne arithmétique. Si cette moyenne excède les 50 %, on estime alors que l'on sort du périmètre de validité de l'approche, et un message d'alarme est émis.*

Problème du bi-matériau :

1er cas : On a un bi-matériau mais la pointe de fissure est dans un seul matériau, cf. Figure 3.1-a. Si on est assuré que la couronne, définie entre les rayons inférieur R_INF et supérieur R_SUP, a comme support des éléments du même matériau, le calcul est possible quelle que soit l'option choisie. Sinon seule l'option 'CALC_G' est possible.

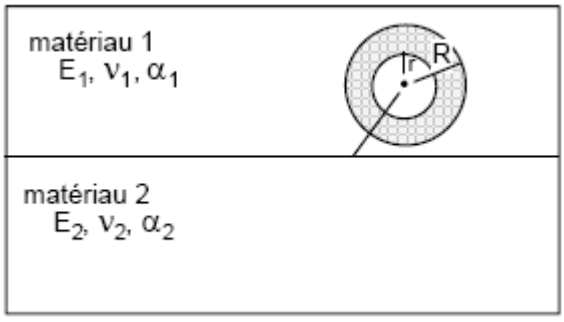
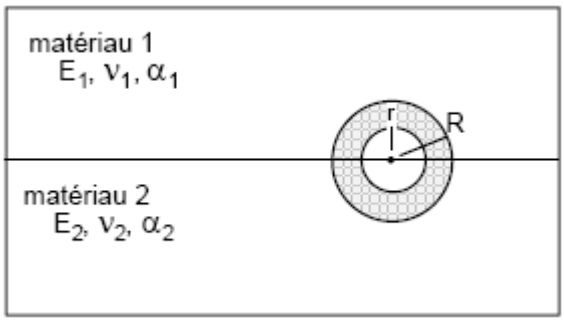


Figure 3.1-a : Bi-matériau : 1er cas

2nd cas : On a un bi-matériau où la pointe de fissure est à l'interface, cf. Figure 3.1-b. À ce jour, seule l'option de calcul du taux de restitution d'énergie (option 'CALC_G') est disponible. Le calcul de coefficients d'intensité de contraintes n'est pas possible dans ce cas.



Document diffusé sous licence GNU FDL

Figure 3.1-b : Bi-matériau : 2 nd cas

Calcul de facteurs d'intensité des contraintes équivalents d'un modèle avec forces cohésives : Pour les études de propagation de fissures en présence de forces cohésives, on peut être amené à calculer des facteurs d'intensité des contraintes équivalents, selon une procédure bien spécifique d'intégrales surfaciques sur la zone cohésive, détaillée dans la documentation de référence [R7.02.19]. Quelques aménagements de syntaxe sont alors nécessaires :

– à l'heure actuelle, la fonctionnalité n'est ouverte que pour les fissures non maillées : le mot-clé FISSURE doit être renseigné, le type de fissure permettant automatiquement de savoir si des forces cohésives sont présentes ou non dans le modèle ;
– les mots-clés R_INF / R_SUP ne doivent pas être renseignés, car il n'est pas nécessaire de définir un tore pour le champ θ ;
– le mot-clé COMPORTEMENT ne doit pas être renseigné, toutes les informations sont extraites du résultat ;
– seuls les lissages LAGRANGE et LAGRANGE_NO_NO sont disponibles ;
– seule l'option CALC_K_G est disponible.
Une syntaxe type est proposée à la fin de cette documentation.

3.2 Opérandes TOUT_ORDRE / NUME_ORDRE / LIST_ORDRE / INST / LIST_INST / TOUT_MODE / NUME_MODE / LIST_MODE / FREQ / LIST_FREQ /PRECISION / CRITERE

Ces opérandes sont utilisés avec l'opérande RESULTAT.

Les opérandes TOUT_ORDRE, NUME_ORDRE, LIST_ORDRE, INST, LIST_INST sont associés aux résultats de type evol_elas, evol_noli ou dyna_trans. Voir [U4.71.00].

Les opérandes TOUT_MODE, NUME_MODE, LIST_MODE, FREQ, LIST_FREQ sont associés aux résultats de type mode_meca.

3.3 Mot clé EXCIT et opérandes CHARGE/FONC_MULT

```
◊  EXCIT = _F(  ♦  CHARGE =   charge
                ◊  FONC_MULT = fmult )
```

Le mot clé EXCIT permet de récupérer une liste de chargements charge, issus des commandes AFFE_CHAR_MECA ou AFFE_CHAR_MECA_F [U4.44.01], et les éventuels coefficients multiplicateurs associés fmult.

Le mot clé EXCIT est facultatif et ne doit pas être renseigné dans le cas général.

Si le mot clé EXCIT est absent de la commande, le chargement pris en compte est celui extrait de resu. Si le chargement est fourni via EXCIT, alors c'est ce chargement qui sera utilisé dans CALC_G. Si le chargement fourni dans EXCIT est différent de celui présent dans resu (cohérence du nom et du nombre de charges, des couples charge-fonction), une alarme est émise et le calcul se poursuit avec les chargements indiqués par l'utilisateur. Attention, cet usage n'est valide que lorsque le résultat est créé via l'opérateur CREA_RESU. En effet, comme CREA_RESU ne permet pas d'appliquer de charges, EXCIT donne la possibilité de définir le chargement directement dans CALC_G.

Les chargements supportés actuellement par les différentes modélisations et pouvant avoir un sens en mécanique de la rupture sont les suivantes :
- Effort volumique : ROTATION, FORCE_INTERNE, PESANTEUR.
- Effort surfacique sur les lèvres de la fissure : FORCE_CONTOUR (2D), FORCE_FACE (3D), PRES_REP.
- Dilatation thermique : la température est transmise via AFFE_MATERIAU/AFFE_VARC

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

- Pré-déformation : PRE_EPSI (cas fissure maillée uniquement, en uniquement 2D pour toutes les options)

En cas de problème thermo-mécanique, la température est transmise via les propriétés matériaux (AFFE_MATERIAU / AFFE_VARC / EVOL). La dilatation thermique est donc automatiquement prise en compte dans le calcul avec CALC_G.

Remarque :

Les chargements non supportés par une option sont ignorés. A ce jour, les chargements suivants pouvant avoir un sens en mécanique de la rupture ne sont pas traités :
- *FORCE_NODALE*
- *FORCE_ARETE*
- *DDL_IMPO sur les lèvres de la fissure*
- *FACE_IMPO*
- *PRE_EPSI en 3D pour les options CALC_G et CALC_K_G*

Il est important de noter que les seuls chargements pris en compte dans un calcul de mécanique de la rupture avec la méthode θ sont ceux supportés par les éléments à l'intérieur de la couronne, où le champ de vecteurs thêta est non nul (entre R_INF et R_SUP [R7.02.01 §3.3]). **Les seuls types de charge susceptibles d'influencer le calcul de G sont donc les chargements volumiques (pesanteur, rotation), un champ de température non uniforme ou des efforts appliqués sur les lèvres de la fissure.**

Attention :

- *Si plusieurs chargements de même nature (par exemple force volumique) apparaissent dans le calcul, ils sont combinés entre eux pour le post-traitement. Il n'est cependant pas possible à ce jour de faire cette combinaison si des chargements de type FORMULE* sont présents *: le calcul se termine alors en erreur.*
- *On applique aussi une règle d'exclusion lors de la présence simultanée d'un champ de pré-déformations (via 'PRE_EPSI') et d'un champ de contraintes initiales.*
- *Il n'est pas possible à ce jour d'associer une charge définie à partir d'une FORMULE et un coefficient multiplicateur (* FONC_MULT *). Dans ce cas, le calcul se termine en erreur.*
- *Les charges cinématiques (*AFFE_CHAR_CINE *et* AFFE_CHAR_CINE_F*), ne peuvent pas être prises en compte dans le calcul.*
- *Pour l'option* CALC_K_G*, si un chargement est imposé sur les lèvres de la fissure (*PRES_REP *ou* FORCE_CONTOUR*), alors il faut* ***obligatoirement*** *orienter correctement les mailles de celles-ci (en utilisant* ORIE_PEAU_2D *ou* ORIE_PEAU_3D *) préalablement au calcul de K (cas fissure maillée uniquement).*
- *Si on fait un calcul en grandes transformations (mot clé* DEFORMATION = 'GROT_GDEP' *ou* 'PETIT_REAC'*) les chargements supportés à proximité du front de fissure (plus exactement dans le tore d'intégration) doivent être des charges mortes, typiquement une force imposée et pas une pression [R7.02.03 §2.4] ; ces charges doivent avoir été déclarées comme non suiveuses dans* STAT_NON_LINE*. Néanmoins, des charges suiveuses peuvent être définies loin de la fissure, car elles n'interviennent alors pas dans le calcul de G.*
- *On note que le calcul de* CALC_G *avec la modélisation AXIS n'est pas disponible pour les résultats des calculs thermo -mécaniques en grande déformation et grande rotation.*

3.4 Mot clé THETA

Le champ thêta est calculé dans CALC_G à partir des mots clés R_INF/R_INF_FO, R_SUP/R_SUP_FO, FOND_FISS/FISSURE.

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

Les différents cas sont décrits dans le tableau ci-dessous selon l'option de calcul, la modélisation (2D ou 3D) et le type de fissure (fissure maillée ou non).

	CALC_G	**CALC_K_G**
2D - Fissure maillée	♦ FOND_FISS ♦ R_INF, ..	♦ R_INF,R_SUP... ♦ FOND_FISS
2D - Fissure non maillée	◊ R_INF, R_SUP... ♦ FISSURE	◊ R_INF, R_SUP... ♦ FISSURE
3D - Fissure maillée	◊ R_INF, R_SUP... ♦ FOND_FISS	◊ R_INF, R_SUP... ♦ FOND_FISS
3D - Fissure non maillée	◊ R_INF, R_SUP... ♦ FISSURE	◊ R_INF, R_SUP... ♦ FISSURE

Conseils sur le choix des couronnes (dans CALC_G) :

- *Éviter d'utiliser un champ thêta défini avec un rayon inférieur* R_INF *nul. Les champs de déplacements sont singuliers en fond de fissure et introduisent des résultats imprécis en post-traitement de mécanique de la rupture.*
- *Il est conseillé d'utiliser successivement la commande* CALC_G *avec au moins trois champs thêta de couronnes différentes pour s'assurer de la stabilité des résultats. En cas de variation importante (supérieure à 5-10%) il faut s'interroger sur la bonne prise en compte de toute la modélisation.*
- *Pour l'option* CALC_K_G *en 2D-axisymétrique, le rayon des couronnes doit être petit devant le rayon du fond de fissure pour avoir la meilleure précision possible. Il est interdit d'avoir des couronnes de rayon plus grand que le rayon du fond de fissure.*

3.4.1 Opérandes FOND_FISS, FISSURE

♦ / FOND_FISS = ff,
/ FISSURE = fiss,

Ces opérandes permettent de définir le (ou les) champs thêta.
Il y a deux possibilités :

1) Si FOND_FISS est renseigné :
ff est le fond de fissure défini par la commande DEFI_FOND_FISS [U4.82.01] pour un fond de fissure ouvert ou fermé (fonds doubles interdits dans CALC_G).
Ce mot-clé ne peut être utilisé que si la fissure est maillée.

2) Si FISSURE est renseigné :
fiss est la fissure définie par la commande DEFI_FISS_XFEM [U4.82.08].
Ce mot-clé est obligatoire si la fissure n'est pas maillée.

Ce mot-clé est obligatoire si la fissure n'est pas maillée. Dans les quelques cas où on réalise une étude de propagation avec des éléments cohésifs, cette fissure est de type COHESIF (mot-clé TYPE_DISCONTINUITE dans DEFI_FISS_XFEM), et l'opérateur effectue alors automatiquement un calcul de facteurs d'intensité des contraintes équivalents par une méthode spécifique d'intégrales surfaciques sur la zone cohésive, détaillée dans [R7.02.19], §6.3.

3.4.2 Opérandes R_INF, R_INF_FO, R_SUP, R_SUP_FO

Ces opérandes permettent de calculer le champ thêta lorsque celui-ci n'a pas été préalablement déterminé. Ils correspondent respectivement aux rayons inférieur et supérieur des couronnes (scalaire ou fonction, en 3D, de l'abscisse curviligne).

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

Les deux rayons peuvent être introduits soit par des valeurs réelles constantes qui sont arguments des mots clés simples R_INF et R_SUP ; soit par des fonctions de l'abscisse curviligne sur le fond de fissure orienté, qui sont arguments des mots clés simples R_INF_FO et R_SUP_FO.

Quelques conseils sont donnés ci-dessous.
En 3D, lorsque les rayons ne sont pas fonction de l'abscisse curviligne, les opérandes R_INF et R_SUP sont facultatifs. S'ils ne sont pas indiqués, ils sont automatiquement calculés à partir du maximum h des tailles de mailles connectées aux nœuds du fond de fissure. Ces tailles de mailles en chaque nœud du fond sont calculées dans la commande DEFI_FOND_FISS, dans le cas d'une fissure maillée ou DEFI_FISS_XFEM, dans le cas d'une fissure non-maillée, et sont présentes respectivement dans le concept fond_fiss [D4.10.01] ou fiss_xfem [D4.10.01]. Il a été choisi de poser R_SUP = 4h et R_INF = 2h. Si on choisit la valeur automatiquement calculée pour R_SUP et R_INF, il convient toutefois de s'assurer que ces valeurs (affichées dans le fichier .mess) sont cohérentes avec les dimensions de la structure.

Dans le cas d'un défaut initialement ouvert et dont le fond n'est pas plan, il n'est à l'heure actuelle pas possible de calculer le taux de restitution d'énergie.

3.4.3 Opérande NUME_FOND

◊ NUME_FOND = n,

Ce mot clé, facultatif, ne doit être défini que pour des fissures non maillées (modélisation X-FEM), le mot-clé FISSURE étant donc renseigné.
Il peut arriver, pour certaines structures, que le fond de fissure soit discontinu. Dans le cas d'une fissure définie par DEFI_FISS_XFEM le fond de fissure est alors découpé en plusieurs parties.
L'opérande NUME_FOND permet d'indiquer sur laquelle de ces sous-parties du fond de fissure on souhaite réaliser le calcul. Par défaut, le calcul se fait sur le premier fond de fissure.

3.4.4 Opérande NB_POINT_FOND

◊ NB_POINT_FOND = nbnofo,

Ce mot clé, facultatif, peut être défini pour des fissures maillées et non maillées, le mot-clé FOND_FISS ou FISSURE étant donc renseigné.

Par défaut, le calcul se fait sur tous les nœuds du fond de fissure pour une fissure maillée ou tous les points du fond de fissure, c'est-à-dire tous les points d'intersection entre le fond de fissure et les arêtes du maillage, pour une fissure non maillée. Dans le cas FEM, si le maillage est libre, le nombre de nœuds en fond de fissure peut être important, ce qui conduit à des temps de calcul très longs, mais également à de fortes oscillations qui ne permettent pas un post-traitement correct sur les paramètres $G(s)$ ou $K(s)$ calculés pour un lissage de LAGRANGE. Dans le cas X-FEM, les points du fond de fissure peuvent être très irrégulièrement espacés, ce qui peut conduire à des oscillations gênantes sur les paramètres $G(s)$ ou $K(s)$ calculés.

L'opérande NB_POINT_FOND permet de fixer a priori le nombre de points de post-traitement, afin d'améliorer la régularité des résultats. Les $nbnofo$ points sont équi-répartis le long du fond de fissure. Quelques conseils sont donnés dans le §3.7.

Ce mot-clé n'est utilisable que pour un lissage de LAGRANGE-LAGRANGE.

3.5 Mot clé COMPORTEMENT

◊ COMPORTEMENT = _F(

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

Ce mot clé facteur permet de redéfinir le comportement du matériau. Mais l'utilisation de ce mot clé ne **doit pas être systématique** : en effet, par défaut, la loi de comportement utilisée dans CALC_G est identique à celle utilisée pour le calcul mécanique (via MECA_STATIQUE ou STAT_NON_LINE). Le fait de renseigner le mot clef comportement crée une Alarme, mais le calcul continue, il appartient à l'utilisateur de vérifier que les comportements lors de la résolution mécanique et du calcul de G sont identiques.
Le calcul du taux de restitution d'énergie G n'a de sens qu'en **élasticité** linéaire ou non linéaire.
Enfin, la seul variable de commande (voir [U4.43.03] : opérateur AFFE_MATERIAU, mot-clé AFFE_VARC) autorisée pour le calcul du taux de restitution d'énergie est la température 'TEMP'.

Remarques :

- *Rien n'interdit d'affecter un comportement différent lors du calcul des déplacements (par exemple élastoplastique) puis de réaliser ce post-traitement avec une autre relation (par exemple élastique non-linéaire). Une vérification de cohérence est effectuée sur les comportements utilisés pour le calcul et pour le post-traitement, et un message d'alarme est émis s'il y a une différence ; l'utilisateur est responsable de l'interprétation des résultats obtenus [R7.02.03].*
- *Par exemple, si le chargement est parfaitement radial monotone, les calculs en élasticité non linéaire et en élastoplasticité conduisent aux mêmes résultats.*

Pour plus de précisions, se reporter à [U2.05.01].

3.5.1 Opérande RELATION pour les lois de comportement élastiques

- RELATION =

Les relations de comportement élastiques possibles ('ELAS', 'ELAS_VMIS_LINE', 'ELAS_VMIS_TRAC', 'ELAS_VMIS_PUIS') sont détaillées dans [U4.51.11].

/ 'ELAS'

Relation de comportement élastique linéaire c'est-à-dire que la relation entre les déformations et les contraintes considérées est linéaire [R7.02.01 §1.1].

Il est possible de définir un état non nul de contraintes initiales (voir mot clé ETAT_INIT), ce qui conduit à considérer le comportement élastique comme incrémental.

/ 'ELAS_VMIS_LINE'

Relation de comportement élastique non linéaire, de Von Mises à écrouissage isotrope linéaire. Les données matériaux nécessaires du champ matériau sont fournies dans l'opérateur DEFI_MATERIAU (cf. [R7.02.03 §1.1] et [R5.03.20]).

/ 'ELAS_VMIS_TRAC'

Relation de comportement élastique non linéaire, de Von Mises à écrouissage isotrope non linéaire. Les données matériaux nécessaires du champ matériau sont fournies dans l'opérateur DEFI_MATERIAU (cf. [R7.02.03 §1.1] et [R5.03.20]).

/ 'ELAS_VMIS_PUIS'

Relation de comportement élastique non linéaire, de Von Mises à écrouissage isotrope non linéaire défini par une loi puissance. Les données matériaux nécessaires du champ matériau sont fournies dans l'opérateur DEFI_MATERIAU (cf. [R7.02.03 §1.1] et [R5.03.20]).

3.5.2 Opérande ETAT_INIT

◊ ETAT_INIT=_F(SIGM = siefelga)

Dans le cas d'une relation de comportement élastique incrémentale, il est possible de définir un état de contraintes initiales non nul.

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

Seules les options CALC_G et CALC_K_G peuvent prendre en compte cet état de contraintes initiales.
Compte tenu de la difficulté de validation de la formulation implantée, il n'est actuellement pas licite de cumuler une pré-déformation (*via* le mot clé PRE_EPSI de l'opérateur AFFE_CHAR_MECA) et une contrainte initiale.
Le champ de contrainte initiale fourni peut être de type SIEF_ELGA, SIEF_ELNO ou SIEF_NOEU dans une modélisation FEM, uniquement SIEF_ELGA pour une modélisation X-FEM (possibilité de les créer à partir de CREA_CHAMP en particulier).
Dans tous les cas, ce champ de contrainte initiale doit être **auto-équilibré**, en absence de fissure, avec les seules conditions aux limites. L'utilisateur doit vérifier que son champ de contraintes initiales est licite en l'appliquant dans le mot clé ETAT_INIT de l'opérateur STAT_NON_LINE, avec un comportement élastique linéaire (RELATION = 'ELAS'), avec les seules conditions aux limites ; le résultat mécanique doit être le même champ de contrainte sans déformations supplémentaires (voir Figure 3.1).
Les calculs en présence d'un état initial peuvent être effectués avec une fissure maillée (FEM) ou bien une fissure non-maillée (X-FEM).

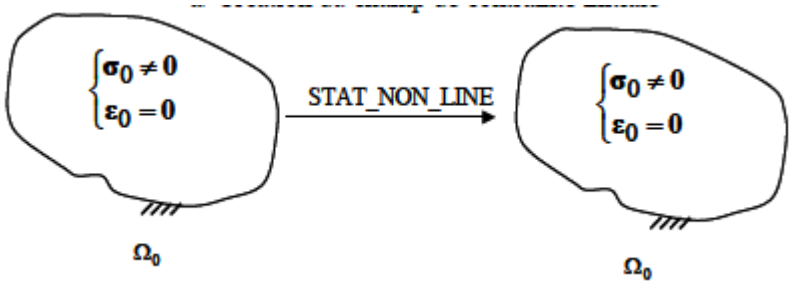


Figure 3.1 : Vérification de la validité du champ de contrainte initiale.

3.5.3 Opérande DEFORMATION

Ce mot-clé permet de définir les hypothèses utilisées pour le calcul des déformations. Pour plus de précisions sur les formalismes de déformations, voir le paragraphe DEFORMATION de [U4.51,11].
Pour commencer, les grandeurs calculées par CALC_G sont définies uniquement en petites déformations. Il n'est donc pas possible d'utiliser pour CALC_G un autre formalisme que PETIT, GROT_GDEP ou PETIT_REAC. Si tel est le cas, le calcul s'arrête en erreur.
Ensuite, il faut en théorie une cohérence entre les formalismes de déformations utilisés pour le calcul mécanique lui-même et le post-traitement dont il est question ici. Cela signifierait donc que les calculs mécaniques eux-mêmes doivent être réalisés avec les formalismes PETIT, GROT_GDEP ou PETIT_REAC uniquement. Toutefois, nous laissons la possibilité de réaliser le post-traitement en petites déformations (PETIT) à partir du résultat d'un calcul mécanique réalisé avec un autre formalisme (par exemple GDEF_LOG). Dans ce cas, une alarme est émise, et charge à l'utilisateur de décider si le résultat fourni a un sens ou non. Pour réaliser un calcul plus fiable du taux de restitution d'énergie en grandes déformations, on conseille d'utiliser la méthode d'équivalence en ouverture présentée succinctement dans U2.05.01.

◊ DEFORMATION =
/ 'PETIT' : les déformations utilisées dans la relation de comportement sont les relations linéarisées. Cela signifie que l'on reste en Hypothèse de Petites Perturbations : petits déplacements, petites rotations et petites déformations. Dans ce cas, les calculs des grandeurs sont licites et validés. Cette option est la seule possible pour les fissures non maillées.

/ 'GROT_GDEP' : Permet de traiter les grandes rotations et les grands déplacements, mais en restant en petites déformations

/ 'PETIT_REAC' : disponible uniquement en comportement incrémental, c'est une approximation des grandes déformations pour laquelle les incréments de déformations sont

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

calculés dans la géométrie actuelle (réactualisée). Elle n'est valable que pour de petits incréments et pour des rotations faibles [U4.51.11].

3.5.4 Opérande CALCUL_CONTRAINTE

```
◊ CALCUL_CONTRAINTE = / 'OUI'                [DEFAUT]
                      / 'NON',
```

Cet opérande est disponible qu'en élasticité (linéaire ou non) et uniquement pour les fissures maillées, sans état initial et pour l'option 'CALC_G'.

Si CALCUL_CONTRAINTE = 'OUI', les contraintes sont recalculées dans l'opérateur CALC_G, à partir du champ de déplacement et de la loi de comportement.

Si CALCUL_CONTRAINTE = 'NON', alors G est calculé sans recalculer les contraintes à partir des déplacements solution (on utilise directement celles présentes dans la structure de données résultat).

Remarque :

Si les lois de comportement utilisées pour le calcul mécanique et pour le post-traitement sont les mêmes - ce qui constitue la pratique normale - alors les résultats avec ou sans recalcul des contraintes sont identiques.
Une pratique usuelle pour prendre en compte la plasticité consiste cependant à faire un calcul mécanique élastoplastique, suivi d'un post-traitement élastique non linéaire pour le calcul de G.
Si on reste bien dans le domaine de validité du calcul de G (chargement radial et monotone), alors les résultats avec ou sans recalcul des contraintes sont identiques. Dès qu'on sort de ce domaine de validité, l'écart croît.
Cette option, à réserver aux utilisateurs avertis, permet donc de vérifier a posteriori qu'on reste bien dans les hypothèses de calcul de G.

3.5.5 Opérandes TOUT / GROUP_MA

```
◊ / TOUT     = 'OUI' ,
  / GROUP_MA = lgrma ,
```

Spécifie les mailles ou les nœuds sur lesquels la relation de comportement est utilisée.

3.5.6 Relation de comportement disponible pour chaque option

		'CALC_G'	'CALC_K_G'
COMPORTEMENT	'ELAS'	'PETIT' 'GROT_GDEP'	'PETIT'
	'ELAS_VMIS_LINE' 'ELAS_VMIS_TRAC' 'ELAS_VMIS_PUIS'	'PETIT' 'GROT_GDEP'	non disp.

Tableau 3.6.4-a : Disponibilité, par option, des relations de comportement.

3.6 Opérande OPTION

```
♦ OPTION = / 'CALC_G' ,
           / 'CALC_K_G',
```

3.6.1 OPTION = 'CALC_G' [R7.02.01] et [R7.02.03]

Elle permet le calcul du taux de restitution de l'énergie G par la méthode thêta en 2D ou en 3D local pour un problème thermo-élastique linéaire ou non linéaire.
En 2D, pour la modélisation AXIS, il faut diviser le résultat obtenu par le rayon en fond de fissure, cf. §4.2.

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

3.6.2 OPTION = 'CALC_K_G' [R7.02.05]

Cette option calcule **en 2D et en 3D** le taux de restitution G et les coefficients d'intensité de contraintes K_1, K_2 et K_3 en thermo-élasticité linéaire plane par la méthode des champs singuliers (utilisation de la forme bilinéaire de G, [R7.02.05]).

Remarques :

- *Pour cette option, seuls les calculs élastiques linéaires sans état initial ou élastique linéaire avec contrainte initiale sont disponibles.*
- *Pour cette option en 2D, si* INFO *vaut 2, on génère le calcul et l'impression (dans le fichier* MESSAGE *) de l'angle de propagation de la fissure. Cet angle, calculé selon 3 critères ($K1$ ou G maximal, $K2$ minimal) d'après les formules d'AMESTOY, BUI et DANG-VAN [R7.02.05 §2.5], est donné à 10 degrés près.*
- *Dans le cas où la fissure est maillée (c'est-à-dire si le mot-clé* FOND_FISS *est renseigné), le calcul de cette option n'est possible que si les lèvres sont initialement collées (* CONFIG_INIT='COLLEE' *dans* DEFI_FOND_FISS [U4.82.01] *).*

3.7 Mot clé LISSAGE

Le domaine d'application de ce mot clé se limite au cas 3D local.

3.7.1 Opérande LISSAGE_THETA

```
◊   LISSAGE_THETA = /   'LEGENDRE'          [DEFAUT]
                    /   'LAGRANGE'
```

La trace du champ thêta sur le fond de fissure peut être discrétisée soit suivant la base des N premiers polynômes de Legendre ('LEGENDRE'), soit suivant les fonctions de forme associées à la discrétisation du fond de fissure ('LAGRANGE') [R7.02.01].

LISSAGE_THETA ='LEGENDRE' : $\theta(s)$ est discrétisé sur une base de polynômes de Legendre $\gamma_j(s)$ de degré j ($0<j<Deg_{max}$) où Deg_{max} est le degré maximal donné sous le mot clé DEGRE (entre 0 et 7).

LISSAGE_THETA ='LAGRANGE' : $\theta(s)$ est discrétisé sur les fonctions de forme du nœud k du fond de fissure : $\phi_k(s)$.

3.7.2 Opérande LISSAGE_G

```
LISSAGE_G =      /   'LEGENDRE'    ,          [DEFAUT]
                 /   'LAGRANGE'    ,
                 /   'LAGRANGE_NO_NO'
```

$G(s)$ peut être discrétisé soit suivant les polynômes de Legendre ('LEGENDRE'), soit suivant les fonctions de forme des nœuds du fond de fissure ('LAGRANGE'). La méthode 'LAGRANGE_NO_NO' est issue de la méthode LAGRANGE-LAGRANGE mais elle est simplifiée [R7.02.01].

Si le lissage de thêta par polynômes de Legendre a été retenu au mot clé précédent, alors le lissage de G doit lui aussi être de type Legendre.

Les options disponibles dans *Aster* sont résumées dans le tableau suivant :

	Théta

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

$G(s)$	Polynômes de LEGENDRE	Polynômes de LEGENDRE	Fonctions de forme
	Polynômes de LEGENDRE	LISSAGE_THETA= 'LEGENDRE' LISSAGE_G = 'LEGENDRE'	LISSAGE_THETA='LAGRANGE' LISSAGE_G= 'LEGENDRE'
	Fonctions de forme		LISSAGE_THETA = 'LAGRANGE' LISSAGE_G = 'LAGRANGE' ou 'LAGRANGE_NO_NO'

Avec X-FEM, le couple LISSAGE_THETA='LAGRANGE'/LISSAGE_G= 'LEGENDRE' est impossible.

3.7.3 Opérande DEGRE

◊ DEGRE = n

n est le degré maximal des polynômes de Legendre utilisés pour la décomposition du champ θ en fond de fissure [§3.12] (lorsque LISSAGE_THETA ='LEGENDRE').

Par défaut n est affectée à 5. La valeur de n doit être comprise entre 0 et 7.

Si on retient les discrétisations LISSAGE_THETA ='LAGRANGE' et LISSAGE_G = 'LEGENDRE', on doit avoir $n<NNO$, où NNO est le nombre de nœuds en fond de fissure [R7.02.01 §2.3].

Conseils sur le lissage :

- *il est difficile de donner une préférence à l'une ou l'autre méthode de lissage. En principe les deux donnent des résultats numériques équivalents. Néanmoins le lissage de type 'LAGRANGE' est peu plus coûteux en temps CPU que le lissage de type 'LEGENDRE' ;*
- *le lissage de type 'LEGENDRE' est sensible au degré maximal des polynômes choisis. Le degré maximal doit être défini en fonction du nombre de nœuds en fond de fissure NNO. Si n est trop grand au regard de NNO les résultats sont médiocres [U2.05.01 §2.4] ;*
- *des oscillations peuvent apparaître avec le lissage de type 'LAGRANGE', en particulier si le maillage comporte des éléments quadratiques ou si la fissure n'est pas maillée. Si le maillage est rayonnant en fond de fissure (fissure maillée), il est alors recommandé de définir des couronnes R_INF et R_SUP coïncidant avec les frontières des éléments. Un lissage de type 'LAGRANGE_NO_NO' permet de limiter ces oscillations ;*
- *pour les fissures maillées et non maillées, lorsque l'on utilise un lissage de type 'LAGRANGE' il est recommandé d'utiliser l'opérande NB_POINT_FOND pour garantir une équi-répartition des points de calculs en fond de fissure. Le choix d'un rapport de l'ordre de 5 entre le nombre de points total en fond de fissure (à chercher dans les informations imprimées dans le fichier message par la commande DEFI_FOND_FISS ou DEFI_FISS_XFEM) et le nombre de points de calcul semble approprié pour limiter les oscillations ;*
- *l'utilisation d'au moins deux types de lissage avec plusieurs couronnes d'intégration et la comparaison des résultats est **indispensable** afin de vérifier la validité du modèle.*

3.8 Opérande TITRE

◊ TITRE = titre

[U4.03.01].

3.9 Opérande INFO

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

◊ INFO = /1, [DEFAUT]
/2,

Niveau de messages dans le fichier 'MESSAGE'.

3.10 Table produite

La commande CALC_G génère un concept de type table.

Cette table est définie comme suit pour les options CALC_G et CALC_K_G :

	Dans tous les cas	Dépend du calcul réalisé		En 2D et 3D local uniquement			En 3D local		
FEM	NUME_FOND	NUME_CAS	NOM_CAS	NOEUD	COOR_X	COOR_Y	COOR_Z	NUM_PT	ABSC_CURV
		NUME_MODE							
		NUME_ORDR	INST						
X-FEM	NUME_FOND	NUME_CAS	NOM_CAS		COOR_X	COOR_Y	COOR_Z	NUM_PT	ABSC_CURV
		NUME_MODE							
		NUME_ORDR	INST						
THETA		NUME_CAS	NOM_CAS						
		NUME_MODE							
		NUME_ORDR	INST						

Tableau 3.10-1: Tableau obtenu avec CALC_G (1)

	Dans tous les cas	CALC_K_G			CALC_K_G en 3D local
FEM	G	K1	K2	G_IRWIN	K3
X-FEM	G	K1	K2	G_IRWIN	K3
THETA	G	K1	K2	G_IRWIN	K3

Tableau 3.10-2: Tableau obtenu avec CALC_G (2)

G_{IRWIN} , est le taux de restitution de l'énergie obtenu à partir des facteurs d'intensité des contraintes K_1 et K_2 (et K_3) avec les formules suivantes :

$$G_{IRWIN}=\frac{1}{E}(K_I^2+K_{II}^2) \text{ en contraintes planes}$$

$$G_{IRWIN}=\frac{(1-\nu^2)}{E}(K_I^2+K_{II}^2) \text{ en déformations planes et axi-symétrique}$$

$$G_{IRWIN}=\frac{(1-\nu^2)}{E}(K_I^2+K_{II}^2)+\frac{K_{III}^2}{2\mu} \text{ en 3D}$$

avec E module de Young et ν coefficient de Poisson et $\mu=\frac{E}{2(1+\nu)}$. La comparaison entre G et G_{IRWIN} permet de s'assurer de la cohérence des résultats : un écart trop important doit conduire à vérifier les paramètres du calcul (raffinement du maillage, choix des couronnes pour thêta, lissage en 3D...).

La commande IMPR_TABLE [U4.91.03] permet d'imprimer les résultats au format voulu.

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

4 Normalisation du taux de restitution G

4.1 2D contraintes planes et déformations planes

En dimension 2 (contraintes planes et déformations planes), le fond de fissure est réduit à un point et la valeur $G(\theta)$ issue de la commande CALC_G est indépendante du choix du champ θ :

$$G=G(\theta) \quad \forall \theta \in \Theta$$

4.2 Axisymétrie

En axisymétrique il faut normaliser la valeur $G(\theta)$ obtenue avec *Aster* pour l'option CALC_G :

$$G=\frac{1}{R}G(\theta)$$

où R est la distance du fond de fissure à l'axe de symétrie [R7.02.01 §2.4.4].

Pour l'option CALC_K_G, les valeurs de G et de K fournies dans le tableau résultat sont directement les valeurs locales, il ne faut donc pas les normaliser.

4.3 Symétrie du modèle

Si on ne modélise que la moitié du solide par rapport à la fissure :

- soit préciser le mot clé SYME = 'OUI' dans les commandes concernées ;
- soit ne pas oublier de multiplier par 2, les valeurs du taux de restitution d'énergie G ou $G(s)$ et par 4 celles de G_{Irwin}. De plus les valeurs des facteurs d'intensité des contraintes correspondantes au mode de symétrie doivent aussi être multipliées par 2.

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

5 Exemples

5.1 Exemple d'utilisation en 3D

Prenons le cas d'une fissure maillée :

Le fond de fissure est défini dans DEFI_FOND_FISS :

```
ff=DEFI_FOND_FISS(  MAILLAGE=MA,
                    FOND_FISS=_F(GROUP_MA = 'LFF'),
                    )
```

Exemples de calcul du taux de restitution de l'énergie en 3D (local) :

```
G1LOC  =  CALC_G (  OPTION       =  'CALC_G',
                    RESULTAT     =  resu,
                    THETA        = _F(  FOND_FISS    =  ff,
                                        R_INF        =  1.,
                                        R_SUP        =  2.,),
                    LISSAGE      = _F(  LISSAGE_G ='LAGRANGE'
                                        LISSAGE_THETA='LAGRANGE'),)
```

Exemple de calcul des facteurs d'intensité des contraintes en 3D :

```
KLOC   =  CALC_G (  OPTION       =  'CALC_K_G',
                    RESULTAT     =  resu,
                    THETA        = _F(  FOND_FISS    =  ff,
                                        R_INF        =  1.,
                                        R_SUP        =  2.,),
                    LISSAGE      = _F(  LISSAGE_G ='LAGRANGE'
                                        LISSAGE_THETA='LAGRANGE'),)
```

On peut trouver des exemples d'utilisation dans les tests suivants :

SSLV110 [V3.04.110] Fissure semi-elliptique en milieu infini
SSLV112 [V3.04.112] Fissure circulaire en milieu infini
HPLV103 [V7.03.103] Thermoélasticité avec fissure circulaire en milieu infini*

Prenons le cas d'une fissure non maillée, et d'un calcul de facteurs d'intensité des contraintes équivalents en présence de forces cohésives.

La fissure a été définie par :

```
fiss=DEFI_FISS_XFEM(MODELE=MO,
                    TYPE_DISCONTINUITE = 'COHESIF',
                    DEFI_FISS=_F(FONC_LN = fonc1,
                                 FONC_LT = fonc2,
                                 GROUP_MA_BORD = group_ma,),
                     ) ;
```

Après le calcul mécanique ayant donné un résultat res, le post-traitement pour obtenir les facteurs d'intensité des contraintes équivalents se fait par :

```
table_k=CALC_G(OPTION='CALC_K_G',
               RESULTAT = res,
               INST = inst_fin,
               THETA =_F(FISSURE = fiss,
                         NB_POINT_FOND = nb_pts,),
```


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

```
LISSAGE = F(LISSAGE_THETA = 'LAGRANGE',
            LISSAGE_G = 'LAGRANGE_NO_NO'),
)
```


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