

Manuel d'Utilisation
Fascicule U4.8- : Post-traitement et analyses dédiées
Document : U4.82.03

Opérateur CALC_G

1 But

Cet opérateur calcule les grandeurs de mécanique de la rupture suivantes :

- le taux de restitution d'énergie (2D, 3D global ou 3D local) 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],
- le taux de restitution d'énergie avec propagation lagrangienne, (c'est-à-dire pour une extension de la fissure en utilisant le même maillage) en 2D ou en 3D dans le cas d'un problème thermo élastique linéaire [R7.02.04],
- les facteurs d'intensité de contraintes :
 - K1 et K2 en 2D (déformations planes ou contraintes planes) par la méthode des déplacements singuliers dans le cas d'un problème thermo élastique linéaire [R7.02.05],
 - K1(s), K2(s) et K3(s) en 3D, fonction de l'abscisse curviligne du fond de fissure par la méthode *thêta* couplée avec la méthode X-FEM dans le cas d'un problème élastique linéaire [R7.02.12],
- la forme bilinéaire *g*, fonction d'une série de déplacements, telle que $g(u,u)=G(u)$,
- la maximisation de *G* et de *K1* sous des contraintes bornes.

Pour des études mécano fiabilistes d'évaluation de probabilité d'amorçage de la rupture, on calcule en plus du taux de restitution d'énergie *G*, sa dérivée par rapport à une variation de domaine pilotée par une fonction *thêta* idoine [R7.02.01] [R4.03.01]. Cette option se limite aux problèmes thermo élastiques linéaires 2D s'appuyant sur des éléments finis quadratiques.

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`.

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',                            [DEFAULT]
      / 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',                            [DEFAULT]
      / NUME_MODE = l_ordre,                          [l_I]
      / LIST_MODE = lis,                              [listis]
      / FREQ      = l_inst,                            [l_R]
      / LIST_FREQ = l_reel,                            [listr8]

    ♦ | PRECISION = / prec,                            [R]
      |           / 1.0D-6,                            [DEFAULT]
      | CRITERE   = / 'RELATIF',                        [DEFAULT]
      |           / 'ABSOLU',

# Récupération ou création du champ theta
    ♦ THETA =_F(♦ / THETA = theta,                    [cham_no_sdaster]
                / ♦ / R_INF = r,                      [R]
                  / R_INF_FO = rz,                    [fonction]
                ♦ / R_SUP = r,                        [R]
                  / R_SUP_FO = rz,                    [fonction]
                ♦ / MODULE = m,                      [R]
                  / MODULE_FO = mz,                  [fonction]

                ♦ / DIRECTION = (d1,d2,d3),          [l_R]
                  / DIRE_THETA = chamno              [cham_no_sdaster]

                ♦ / FOND_FISS = ff,                  [fond_fiss]
                  / FISSURE = ffx,                  [fiss_xfem]

                ♦ NUME_FOND = / 1,                    [DEFAULT]
                  / n                                [I],

# Chargement
    ♦ EXCIT =_F ( ♦ CHARGE = charge ,                [char_meca]
                  ♦ FONC_MULT = fmult,              [char_cine_meca]
                  [fonction]
                  [formule]
                ),

    ♦ SYME_CHAR = / 'SANS' ,                        [DEFAULT]
                  / 'SYME' ,
                  / 'ANTI' ,

```

Titre : **Opérateur CALC_G**
Auteur(s) : **E. GALENNE**

Date : **10/04/07**
Clé : **U4.82.03-H** Page : **3/28**

Comportement

```

    ◇ / COMP_ELAS =_F (
        ◇ RELATION      = / 'ELAS',           [DEFAULT]
                          / 'ELAS_VMIS_LINE',
                          / 'ELAS_VMIS_TRAC',
        ◇ DEFORMATION   = / 'PETIT',           [DEFAULT]
                          / 'GREEN',
        ◇ / TOUT        = 'OUI',               [DEFAULT]
          / | GROUP_MA  = lgrma,               [l_gr_maille]
          / | MAILLE    = lma ,                [l_maille]
          / |           ),
    / COMP_INCR =_F (
        ◇ RELATION      = / 'ELAS',           [DEFAULT]
                          / 'VMIS_ISOT_TRAC',
                          / 'VMIS_ISOT_LINE',
        ◇ DEFORMATION   = / 'PETIT',           [DEFAULT]
                          / 'PETIT_REAC' ,
        ◇ / TOUT        = 'OUI' ,              [DEFAULT]
          / | GROUP_MA  = lgrma ,              [l_gr_maille]
          / | MAILLE    = lma ,                [l_maille]
          / |           ),
    ◇ ETAT_INIT =_F (
        ◆ / | DEPL      = dep ,                [cham_no_sdaster]
          / | SIGM      = sig ,                / [carte_SIEF_R]
          / |           / [cham_elem_SIEF_R]
          / |           ),
    ),
```

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

```

    ◇ LISSAGE = _F( ◇ / LISSAGE_THETA = / 'LEGENDRE'      [DEFAULT]
                  / 'LAGRANGE'
                  / 'LAGRANGE_REGU'
                  / LISSAGE_G      = / 'LEGENDRE'      [DEFAULT]
                  / 'LAGRANGE'
                  / 'LAGRANGE_NO_NO'
                  / 'LAGRANGE_REGU'
    ◇ DEGRE =      / 0,
                  / 1,
                  / 2,
                  / 3,
                  / 4,
                  / 5,      [DEFAULT]
                  / 6,
                  / 7,
    ),
```

Option demandée

```

    ◆ OPTION = / 'CALC_G' ,
               / 'CALC_G_GLOB' ,
               / 'CALC_K_G' ,
               / 'K_G_MODA' ,
               / 'G_MAX' ,
               / 'G_MAX_GLOB' ,
               / 'CALC_K_MAX' ,
               / 'G_BILI' ,
               / 'G_BILI_GLOB' ,
               / 'G_LAGR' ,
               / 'G_LAGR_GLOB' ,
```

Titre : *Opérateur CALC_G*

Date : 10/04/07

Auteur(s) : **E. GALENNE**

Clé : U4.82.03-H

Page : 4/28

```
# Si OPTION = 'G_MAX' ou 'G_MAX_GLOB'
  ◇ BORNES =_F(
    ◆ NUME_ORDRE = num , [I]
    ◆ VALE_MIN = qmin , [R]
    ◆ VALE_MAX = qmax , [R]
  ),
# Si OPTION = 'CALC_K_MAX'
  ◇ SIGNES =_F(
    ◆ CHARGE_S = listS , [l_I]
    ◆ CHARGE_NS = listNS , [l_I]
  ),
# Si OPTION = 'G_LAGR' ou 'G_LAGR_GLOB'
  ◇ PROPAGATION = alpha , [R]
  ◇ THETA_LAGR = theta , [theta_geom]
  ◇ DIRE_THETA_LAGR = dire_theta , [cham_no_DEPL_R]

  ◇ SENSIBILITE = (... voir [U4.50.02])
  ◇ TITRE = titre , [l_Kn]

# Impression d'informations

  ◇ INFO = / 1 , [DEFAULT]
           / 2 ,
)
```

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	Sensibilité
D_PLAN	CALC_G G_MAX G_BILI G_LAGR	CALC_K_G K_G_MODA	disponible
C_PLAN	CALC_G G_MAX G_BILI G_LAGR	CALC_K_G K_G_MODA	disponible
AXIS	CALC_G G_MAX G_BILI G_LAGR	/	disponible
3D	CALC_G / CALC_G_GLOB G_MAX / G_MAX_GLOB G_BILI / G_BILI_GLOB G_LAGR / G_LAGR_GLOB	CALC_K_G K_G_MODA CALC_K_MAX	Non disponible

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

Remarques sur le calcul de sensibilité :

- *Le calcul de la dérivée du taux de restitution d'énergie par rapport à une variation de domaine n'est licite que pour les modélisations 2D (`D_PLAN`, `AXIS` et `C_PLAN`) en thermo élasticité linéaire, avec des éléments quadratiques.*
- *Avec cette option, la configuration contraintes planes n'est d'ailleurs prise en compte qu'en post-traitement du calcul de mécanique, c'est-à-dire pour la détermination des tenseurs des déformations et des contraintes à partir des déplacements. Elle ne doit pas apparaître lors du calcul de sensibilité de `MECA_STATIQUE` qui ne supporte que les modélisations `D_PLAN` et `AXIS`. Dans une telle configuration l'utilisateur est bien sûr seul juge de la pertinence de ses résultats.*

Remarques sur les propriétés matériau :

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`', `CALC_G_GLOB`'). Elles doivent être indépendantes de la température pour le calcul des facteurs d'intensité de contraintes.

Le calcul de sensibilité n'a été développé que pour des matériaux élastiques indépendants de la température. Ils peuvent par contre être hétérogènes.

Les caractéristiques SY et D_SIGM_EPSI ne sont traitées que pour un problème élastique non linéaire avec écrouissage de Von Mises et avec l'option de calcul du taux de restitution d'énergie 'CALC_G_GLOB'. 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 AFPE = _F (TOUT = 'OUI') dans la commande AFPE_MATERIAU [U4.43.03], quitte à utiliser la règle de surcharge ensuite.

Problème du bi-matériau :

1^{er} 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 les options 'CALC_G' et 'CALC_G_GLOB' sont possibles.

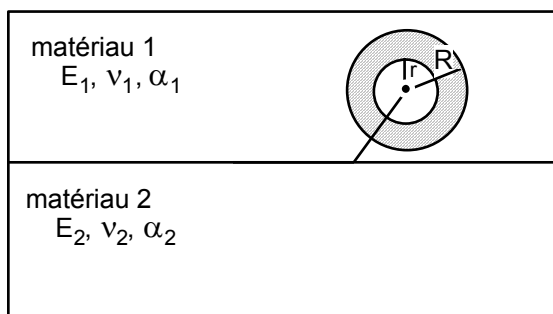


Figure 3.1-a : Bi-matériau : 1^{er} cas

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

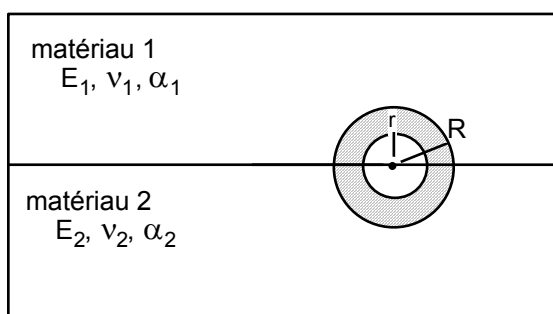


Figure 3.1-b : Bi-matériau : 2nd cas

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. Voir [U4.71.00].

3.3 Mot clé THETA

Le champ `thêta` est :

- soit calculé préalablement par l'opérateur `CALC_THETA` puis transmis par le mot clé facteur `THETA` (cf §3.3.1),
- soit calculé dans `CALC_G` à partir des mots clés `R_INF/R_INF_FO`, `R_SUP/R_SUP_FO`, `MODULE/MODULE_FO`, `FOND_FISS/FISSURE`.

Remarques :

- Dans le cas où on demande le calcul de `thêta`, on peut se dispenser des mots clés `MODULE/MODULE_FO` en 3D local pour les options autres que `G_LAGR` car ils sont calculés automatiquement par Code Aster.
- Pour le calcul de `G` ou `K` local en 3D, le champ `thêta` est nécessairement calculé dans la commande `CALC_G`.

3.3.1 Opérande THETA

♦ / `THETA` = `theta`

Le champ `thêta` est un champ de vecteur en chaque nœud du maillage. C'est un concept de type `cham_no_sdaster`. Il peut être affecté directement avec la commande `AFFE_CHAM_NO` [U4.44.11]. Dans la pratique, il est généralement issu de la commande spécifique `CALC_THETA` [U4.82.02] qui permet d'affecter le module, la direction du champ `thêta` et les rayons de la couronne entourant le fond de fissure.

Pour plus de précisions se reporter à [R7.02.01 §3].

Conseils :

- Eviter d'utiliser un champ `thêta` défini avec un rayon inférieur 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 3 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.
- En 2D, ce champ `thêta` permettant de cerner la zone de calcul autour de la fissure est complètement indépendant du champ `thêta` lié au calcul de sensibilité. L'option prend en compte leurs éventuels recouvrements de supports, voire le déplacement de l'un par rapport à l'autre.

3.3.2 Opérandes `R_INF`, `R_INF_FO`, `R_SUP`, `R_SUP_FO`, `MODULE`, `MODULE_FO`, `DIRECTION`, `DIRE_THETA`

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), au module du champ `thêta` et à sa direction.

L'utilisation de ces opérandes est décrite dans la documentation de `CALC_THETA` [U4.82.02].

3.3.3 Opérandes `FOND_FISS`, `FISSURE`

♦ / `FOND_FISS` = `ff`,

`ff` est le fond de fissure défini par la commande `DEFI_FOND_FISS` [U4.82.01].

- En 3D : Il permet de récupérer :

- la liste ordonnée des nœuds du fond de fissure ;
- les mailles des lèvres de la fissure ou la normale à la fissure ;
- les directions de propagation du fond de fissure aux extrémités.

C'est à partir de ces entités que sont calculées automatiquement les abscisses curvilignes *s* et les directions de propagation du fond de fissure en chaque nœud [R7.02.01 §2.2]. Ce mot clé est obligatoire sauf pour les options '*CALC_K_G*' et '*K_G_MODA*'.

- En 2D : il permet de récupérer :
 - le nœud de fond de fissure,
 - la normale à la fissure.

Ce mot clé est obligatoire pour les options '*CALC_K_G*' et '*K_G_MODA*'.

/ FISSURE = *fiss*,

fiss est la fissure définie par la commande *DEFI_FISS_XFEM* [U4.82.08]. Elle permet de récupérer dans le cadre de la méthode X-FEM en 3D :

- la liste ordonnée des points du fond de fissure ;
- les gradients des level-sets ;
- la base locale au fond de fissure ;
- le statut des nœuds (enrichissement ou pas) et les numéros des mailles enrichies.

Ces entités contiennent notamment les abscisses curvilignes *s* et les directions de propagation du fond de fissure en chaque nœud. Ce mot est obligatoire uniquement dans le cas où *OPTION* = '*CALC_K_G*' ou '*K_G_MODA*'.

3.3.4 Opérande *NUME_FOND*

◇ *NUME_FOND* = *n*,

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. Cet opérande n'est utilisé que si la fissure est définie sous l'opérande *FISSURE*.

3.4 Mot clé *EXCIT* et opérands *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 coefficients multiplicateurs *fmult*.

Le mot clé *EXCIT* est facultatif.

Si le mot clé *EXCIT* est absent de la commande, le chargement pris en compte est celui extrait de *resu*. 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.

Dans tous les cas, il faut veiller à ce que les charges indiquées ici aient bien été prises en compte dans le calcul mécanique précédent qui a produit le champ de déplacements.

Les chargements supportés actuellement par les différentes modélisations et pouvant avoir un sens en mécanique de la rupture sont les suivantes :

- *TEMP_CALCULEE*
- *ROTATION*
- *FORCE_INTERNE*

Titre : Opérateur CALC_G
Auteur(s) : E. GALENNE

Date : 10/04/07
Clé : U4.82.03-H Page : 9/28

- PESANTEUR
- FORCE_CONTOUR (2D) ou FORCE_FACE (3D)
- PRES_REP
- EPSI_INIT (en 2D pour toutes les options, et en 3D uniquement pour l'option CALC_G_GLOB)

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*
- *EPSI_INIT* *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 θ 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 :

- *Un chargement de même nature (par exemple force volumique) ne peut figurer que dans une seule charge. Dans le cas contraire, le calcul se termine en erreur.*
- *On applique aussi une règle d'exclusion lors de la présence simultanée d'un champ de déformations (via 'EPSI_INIT') et de déplacements initiaux (via 'ETAT_INIT/DEPL' cf. [§3.8]). Seul un des deux doit subsister.*
- *Il n'est pas possible à ce jour d'associer une charge définie comme une fonction (AFFE_CHAR_MECA_F) 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.*
- *Si on fait un calcul en grandes transformations (mot clé DEFORMATION : 'GREEN' sous le mot clé facteur COMP_ELAS ou DEFORMATION = 'PETIT_REAC' sous le mot clé facteur COMP_INCR) les chargements supportés 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 suivieuses dans STAT_NON_LINE.*

Lors du calcul de la dérivée de G par rapport à une variation de domaine (calcul de sensibilité), seuls les chargements PRES_REP et TEMP_CALCULEE sont utilisables dans la totalité du processus. Cette restriction logicielle n'est due qu'au développement limité de l'option SENSIBILITE dans l'opérateur MECA_STATIQUE. Comme pour la modélisation C_PLAN, les autres types de chargement ne sont pris en compte qu'en post-traitement du calcul de mécanique. **Ils ne peuvent et ils ne doivent intervenir** que pour l'assemblage des termes de la dérivée. Ils sont donc modélisés par des AFFE_CHAR_MECA ou AFFE_CHAR_MECA_F insérés entre MECA_STATIQUE et CALC_G.

D'autre part on ne peut, pour l'instant, que manipuler des chargements indépendants de la variation de domaine, dans leurs définitions intrinsèques comme dans celles de leurs supports. En d'autres termes, leurs dérivées eulériennes doivent être nulles.

3.5 Opérande SYME_CHAR

◇ SYME_CHAR = / 'SANS' , [DEFAULT]
/ 'SYME' ;
/ 'ANTI' ;

Le mot clé SYME_CHAR permet d'indiquer si le chargement est symétrique ou antisymétrique dans le cas où on ne modélise que la moitié du solide par rapport à la fissure.

Il multiplie par 2 les valeurs du taux de restitution d'énergie G et sa dérivée éventuelle

Si l'option 'CALC_K_G' est utilisée, alors il affecte 0 à K2 si le chargement est symétrique par rapport à la fissure, ou 0 à K1 s'il est antisymétrique.

3.6 Mot clé COMP_ELAS

◇ COMP_ELAS =

Ce mot clé facteur permet de définir la relation de comportement du matériau utilisée.

Par défaut la relation de comportement est élastique linéaire en petites déformations.

Le calcul de la dérivée de G par rapport à une variation de domaine (effectué si SENSIBILITE est renseigné) est restreint à l'élasticité linéaire (en pré et post-traitement), par contre il a été aussi étendu aux déformations de Green-Lagrange.

Remarques :

- *Le calcul du taux de restitution d'énergie G n'a de sens qu'en élasticité linéaire ou non linéaire (COMP_ELAS). Il est cependant possible de calculer en élastoplasticité (COMP_INCR) un paramètre G défini alors comme le flux d'énergie total (plasticité et rupture) à travers le défaut. Dans le cas de l'élastoplasticité, le défaut doit être modélisé par une entaille.*
- *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). L'utilisateur est responsable de l'interprétation des résultats obtenus [R7.02.03].*
- *Si le chargement est parfaitement radial monotone, les calculs en élasticité non linéaire et en élastoplasticité conduisent aux mêmes résultats. Pour ce type de chargement (et uniquement dans ce cas), il est également possible de faire un calcul élastoplastique sur une fissure.*

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

3.6.1 Opérande RELATION

◇ RELATION =

/ '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].

/ '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. l'opérateur STAT_NON_LINE [U4.51.03] et le mot clé VMIS_ISOT_LINE) [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. l'opérateur STAT_NON_LINE [U4.51.03] et le mot clé VMIS_ISOT_TRAC) [R7.02.03 §1.1] et [R5.03.20].

3.6.2 Opérande DEFORMATION

◇ DEFORMATION = / 'PETIT'

Les déformations utilisées dans la relation de comportement sont les relations linéarisées :

$$\varepsilon_{ij}(\mathbf{u}) = \frac{1}{2} (u_{i,j} + u_{j,i})$$

/ 'GREEN'

Les déformations utilisées dans la relation de comportement sont les déformations de Green-Lagrange [R7.02.03 §2.1] :

$$\varepsilon_{ij}(\mathbf{u}) = \frac{1}{2} (u_{i,j} + u_{j,i} + u_{k,i} u_{k,j})$$

Attention :

- Les chargements supportés en grand déplacement sont ceux supportés en élastique linéaire à condition que ce soient des charges mortes : charge imposée ou pression non suiveuse.
- Les déplacements et les rotations peuvent être grands mais il est préférable de se limiter à de petites déformations si l'on souhaite une cohérence avec le matériau réel. Pour plus de précisions se référer à [R7.02.03 §2.5].

3.6.3 Opérandes TOUT / GROUP_MA / MAILLE

◇ / TOUT = 'OUI' ,
/ | GROUP_MA = lgrma ,
| MAILLE = lma ,

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

3.6.4 Relation de comportement disponible pour chaque option

		'CALC_G'	'CALC_K_G', 'K_G_MODA'
COMP_ELAS	'ELAS'	'PETIT'	'PETIT'
		'GREEN'	
	'ELAS_VMIS_LINE'	'PETIT'	non disp.
		'GREEN'	
	'ELAS_VMIS_TRAC'	'PETIT'	non disp.
		'GREEN'	

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

Il est possible pour ces relations de comportement de calculer le taux de restitution d'énergie G en grandes transformations [R7.02.03 §2] à condition d'avoir uniquement des charges mortes.

3.7 Mot clé COMP_INCR

◇ COMP_INCR =

La relation de comportement est élastoplastique associée à un critère de Von Mises avec écrouissage isotrope ou cinématique.

◇ RELATION =

Titre : *Opérateur CALC_G*
Auteur(s) : **E. GALENNE**

Date : 10/04/07
Clé : U4.82.03-H Page : 12/28

/ 'ELAS'

Relation de comportement élastique incrémentale [U4.51.03].

/ 'VMIS_ISOT_LINE'

Von Mises avec écrouissage isotrope linéaire ([U4.51.03] et [R5.03.20]).

/ 'VMIS_ISOT_TRAC'

Von Mises avec écrouissage isotrope donné par une courbe de traction [U4.51.03].

◇ DEFORMATION =

/ 'PETIT'

Déformations linéarisées : $\Delta \varepsilon_{ij} = \varepsilon_{ij}(\Delta u) = 1/2 (\Delta u_{i,j} + \Delta u_{j,i})$

/ 'PETIT_REAC'

$$\Delta \varepsilon_{ij} = 1/2 \left(\frac{\partial \Delta u_i}{\partial (X+u)_j} + \frac{\partial \Delta u_j}{\partial (X+u)_i} \right) \text{ [U4.51.03].}$$

◇ TOUT / GROUP_MA / MAILLE

Spécifient les mailles sur lesquels la relation de comportement incrémentale est utilisée.

3.8 Mot clé ETAT_INIT

◇ ETAT_INIT =

Etat initial de référence choisi. Par défaut, tous les champs sont identiquement nuls. La donnée d'un état initial n'a de sens (et n'est donc prise en compte) que pour la partie du domaine traitée en comportement incrémental (COMP_INCR) : si le calcul est élastique (COMP_ELAS) cela n'a aucune incidence.

Compte tenu de la formule implantée dans le source CALC_G, il n'est pas licite de cumuler une déformation initiale (soit directement dans la charge avec EPSI_INIT, soit comme ci-dessous, sous forme de déplacement), avec un champ de contraintes initiales.

Attention :

- Si l'on veut prendre en compte un état initial en élasticité, c'est le mot clé ELAS situé sous COMP_INCR qu'il faut utiliser.
- On applique une règle d'exclusion lors de la présence simultanée d'un champ de déplacements (via CHARGE/EPSI_INIT cf. [§3.4]) et de déformations initiaux (via ETAT_INIT/DEPL). Seul un des deux doit subsister.

Lors du calcul de la dérivée comme pour les chargements, ces états initiaux ne sont pris en compte qu'en post-traitement du calcul de mécanique. **Ils ne peuvent et ils ne doivent intervenir** que pour l'assemblage des termes de cette dérivée.

D'autre part on ne peut, pour l'instant, que manipuler des états indépendants de la variation de domaine, dans leurs définitions intrinsèques comme dans celles de leurs supports. En d'autres termes, leurs dérivées eulériennes doivent être nulles.

/ SIGM = sig,
/ DEPL = depl,

Respectivement, champs de contraintes et de déplacements pris à l'état initial. Ils peuvent par exemple être issus de la commande RECU_CHAMP, ou bien avoir été lus dans un fichier au format I-DEAS par la commande LIRE_RESU. Soit on donne un déplacement initial, soit une contrainte initiale. Attention, si la charge transmise dans l'opérande CHARGE contient une déformation initiale (mot clé EPSI_INIT de AFFE_CHAR_MECA_F), celle-ci sera prise en compte de la même façon que le déplacement depl fourni ici ; il est alors illicite de donner un état initial avec le mot clé DEPL.

3.9 Opérande OPTION

```

♦  OPTION = / 'CALC_G' ,
           / 'CALC_G_GLOB' ,
           / 'CALC_K_G' ,
           / 'K_G_MODA' ,
           / 'G_MAX' ,
           / 'G_MAX_GLOB' ,
           / 'CALC_K_MAX' ,
           / 'G_BILI' ,
           / 'G_BILI_GLOB' ,
           / 'G_LAGR' ,
           / 'G_LAGR_GLOB' ,

```

3.9.1 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.

En 3D, $G(s)$ est solution de l'équation variationnelle [R7.02.01 §2.2].

$$\int_{\Gamma_0} G(s) \theta(s) \cdot \mathbf{m}(s) ds = \mathcal{G}(\theta) \quad , \quad \forall \theta \in \Theta$$

où Γ_0 est le fond de fissure et \mathbf{m} la normale au fond de fissure dans le plan tangent de ses lèvres.

3.9.2 OPTION = 'CALC_G_GLOB' [R7.02.01] et [R7.02.03]

Elle permet le calcul du taux de restitution de l'énergie G par la méthode *thêta* en 3D global pour un problème thermo élastique linéaire ou non linéaire. Il faut diviser la valeur brute de G donnée par *Aster* par la longueur de la fissure, cf. §4.3.

3.9.3 OPTION = 'CALC_K_G' [R7.02.05]

Cette option calcule **en 2D** le taux de restitution G classique et les coefficients d'intensité de contraintes K_1 et K_2 en thermo élasticité linéaire plane (modélisations *C_PLAN* et *D_PLAN*) par la méthode des champs singuliers (utilisation de la forme bilinéaire de G).

Cette option calcule **en 3D** par la méthode *thêta* le taux de restitution d'énergie $G(s)$ et les facteurs d'intensité des contraintes $K1(s), K2(s), K3(s)$ pour un problème élastique linéaire. C'est une généralisation du cas 2D [R7.02.05].

Les facteurs d'intensité des contraintes sont calculés à partir de la forme bilinéaire symétrique du taux de restitution. Cette formulation utilise les expressions explicites des champs de déplacements singuliers connus pour une fissure plane à fond droit dans un milieu infini. Les déplacements singuliers sont calculés en se plaçant dans l'hypothèse des déformations planes.

Remarques :

- Pour cette option en 3D, seuls les calculs élastiques linéaires sans état initial sont disponibles à ce jour ; la présence du mot clé *FISSURE* est obligatoire.
- 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.

3.9.4 OPTION = 'K_G_MODA' [R7.02.05]

Cette option permet le calcul des coefficients d'intensité de contraintes modaux, i.e. les facteurs d'intensité des contraintes associés aux modes propres de vibration de la structure. Le calcul est possible en 2D (modélisation C_PLAN ou D_PLAN) et en 3D.

Les calculs sont réalisés en thermo élasticité linéaire par la méthode des champs singuliers (utilisation de la forme bilinéaire de G), à partir d'une structure de données RESULTAT de type mode_meca uniquement. Le taux de restitution d'énergie G est aussi calculé.

3.9.5 OPTION = 'G_LAGR' OU 'G_LAGR_GLOB' [R7.02.04]

Ces options concernent la propagation Lagrangienne [R7.02.04] pour les cas 2D ('G_LAGR'), 3D local ('G_LAGR'), et 3D global ('G_LAGR_GLOB'). Il faut fournir la valeur de la propagation derrière le mot clé PROPAGATION.

3.9.6 OPTION = 'G_BILI' OU 'G_BILI_GLOB' [R7.02.01]

Pour une série de déplacements (U_1, \dots, U_n) en 2D ou 3D, cette option permet le calcul de la forme bilinéaire $g(U_i, U_j)$ pour $i \geq j$; si $i = j$ alors $g(u, u) = G(u)$. Les résultats sont stockés dans une table comportant deux indices i et j en référence aux déplacements U_i et U_j ordonnés dans la liste contenue dans la structure de données résultat sous le mot clé RESULTAT.

L'option 'G_BILI' correspond aux calculs 2D et 3D local; l'option 'G_BILI_GLOB' correspond au calcul du taux de restitution d'énergie 3D global.

Attention :

- En 3D local, seules les combinaisons de discrétisation de $G(s)$ et du champ θ , cf. [§3.11] : *LEGENDRE-LEGENDRE* ou *LAGRANGE-LAGRANGE* sont disponibles pour cette option.
- Cette option de calcul n'est valable que pour des calculs élastiques linéaires où la superposition de chargement par combinaison linéaire est possible.

3.9.7 OPTION = 'G_MAX' OU 'G_MAX_GLOB' [R7.02.05]

Cette option concerne uniquement la maximisation de G en 2D ou 3D local (option 'G_MAX') ou 3D global ('G_MAX_GLOB') sous des contraintes bornes [R7.02.05]. Il faut fournir la valeur des contraintes bornes derrière le mot clé BORNES, cf. § 3.12. Attention, cette option ne permet pas de distinguer les chargements conduisant à une ouverture ou à une fermeture de la fissure, contrairement à l'option CALC_K_MAX.

3.9.8 OPTION = 'CALC_K_MAX'

Cette option concerne uniquement la maximisation de K1 en 3D local en présence de chargements signés et non signés. Il faut fournir le signe des chargements derrière le mot clé SIGNES, cf. §3.13. Contrairement aux options G_MAX et G_MAX_GLOB, cette option permet bien de distinguer ouverture et fermeture de la fissure.

3.10 Mot clé LISSAGE

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

3.10.1 Opérande LISSAGE_THETA

◇ / LISSAGE_THETA = / 'LEGENDRE' [DEFAULT]
/ 'LAGRANGE'

/ 'LAGRANGE_REGU'

La trace du champ θ 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' ou 'LAGRANGE_REGU') [R7.02.01].

LISSAGE_THETA = 'LEGENDRE' : $\theta(s)$ est discrétisé sur une base de polynômes de Legendre $\gamma_j(s)$ de degré j ($0 \leq j \leq \text{Deg}_{\max}$) 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 : $\varphi_k(s)$.

LISSAGE_THETA = 'LAGRANGE_REGU' : forme régularisée du lissage de LAGRANGE, consistant à prendre systématiquement des fonctions de forme linéaires et à étendre le support de chaque champ θ sur 4 mailles consécutives (contre 2 avec 'LAGRANGE').

3.10.2 Opérande LISSAGE_G

```
/ LISSAGE_G = / 'LEGENDRE' , [DEFAULT]
/ 'LAGRANGE' ,
/ 'LAGRANGE_NO_NO' ,
/ 'LAGRANGE_REGU'
```

$G(s)$ 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]. De même, 'LAGRANGE_REGU' est une version régularisée de la méthode LAGRANGE, qui peut être utile pour les fonds de fissure maillés irrégulièrement ou avec des éléments quadratiques.

Si le lissage de θ par polynômes de Legendre a été retenu au mot clé précédent, alors le lissage de G doit lui aussi être de type Legendre. De même, si un lissage de θ de type 'LAGRANGE_REGU' a été retenu au mot clé précédent, alors le lissage de G doit lui aussi être de type 'LAGRANGE_REGU'.

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

Théta		
	Polynômes de LEGENDRE	Fonctions de forme
$G(s)$	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' LISSAGE_THETA = 'LAGRANGE_REGU' LISSAGE_G = 'LAGRANGE_REGU'

3.10.3 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 $n \leq NNO$, 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 la méthode Thêta : Lagrange est un peu plus coûteuse en temps CPU que la méthode Thêta : Legendre ;*
- *la méthode Thêta : 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 la méthode Thêta : Lagrange, en particulier si le maillage comporte des éléments quadratiques. Si le maillage est rayonnant en fond de fissure, 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`' ou '`LAGRANGE_REGU`' permet de limiter ces oscillations ;*
- *l'utilisation d'au moins deux types de lissage avec plusieurs couronnes d'intégration et la comparaison des résultats permet de conforter la validité du modèle.*

3.11 Opérande SENSIBILITE

◆ `SENSIBILITE = thêta`

Nom du paramètre sensible par rapport auquel on dérive (voir [U4.50.02]).

Avec cette opérande on a accès, en plus de la valeur du taux de restitution d'énergie telle qu'elle est fournie avec '`CALC_G`', à sa dérivée par rapport à une variation de domaine décrite par le champ `thêta` sensibilité.

Son périmètre d'application se limite aux **calculs thermo élastiques linéaires 2D**, s'appuyant sur des **éléments finis quadratiques** complets ou incomplets (`SEG3`, `TRIA6`, `QUAD8` et `QUAD9`). Elles supportent divers modélisations (cf [§3.1]), chargements (cf [§3.4]) et états initiaux (cf [§3.8]) en pré ou post-traitements du calcul mécanique. Les matériaux peuvent être hétérogènes mais ils doivent être indépendants de la température.

Remarque :

Le champ d'investigation de cette option est connexe de celui de l'option '`G_LAGR`'. Dans les deux cas on évite de coûteuses études paramétriques en utilisant un maillage fixe de référence et en modélisant les variations virtuelles de domaine par des fonctions `thêta` appropriées.

Le champ $\theta_s = \text{thêta}$ est un champ de vecteur 2D en chaque nœud du maillage. Il est orienté suivant l'axe des abscisses. C'est un concept de type `cham_no_sdaster`. Il peut être affecté directement avec la commande `AFFE_CHAM_NO` [U4.44.11].

Dans la pratique, il est généralement issu de la commande spécifique `CALC_THETA` [U4.82.02] avec l'option `THETA_BANDE` qui permet de saisir le module (mot-clé `MODULE`) et les abscisses x_1 et x_2 (mot-clé `R_INF` et `R_SUP`) des points délimitant son support vertical. On rappelle que ce champ décroît de la valeur `MODULE` à la valeur nulle entre les abscisses x_1 et x_2 , et qu'il est nul partout ailleurs. Ces abscisses peuvent être négatives mais on doit avoir $x_1 < x_2$.

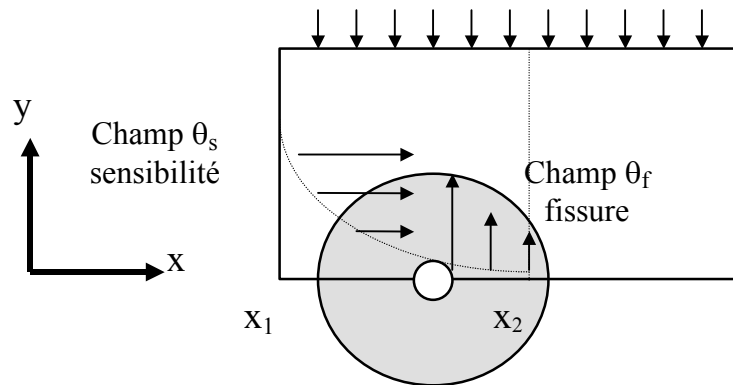


Figure 3.11-a : Dérivée de $G(\theta_f)$ par rapport à une variation de domaine pilotée par θ_s

Remarques :

- Contrairement au champ θ_f fissure qui est juste continu et défini sous forme d'un polynôme du premier ordre, ce champ θ_s est une combinaison de monômes du troisième ordre et il est de classe C^2 sauf au milieu de son support (où il est juste C^1).
- En effet, lors du calcul de G on ne fait appel qu'aux dérivées premières du champ θ_f fissure, alors que pour l'obtention de sa dérivée on utilise les dérivées secondes du θ_s sensibilité. Un compromis a donc été trouvé entre l'ordre théorique requis par les dérivations et la précision des éléments finis modélisant le calcul. Ainsi il faut avoir recours à des éléments quadratiques pour estimer cette dérivée.

3.12 Mot-clé BORNES

◇ BORNES =

Ce mot clé facteur est obligatoire si on utilise les options 'G_MAX' et 'G_MAX_GLOB'. Sinon il n'est pas utilisé. Il permet de définir des couples de contraintes bornes (q_i^-, q_i^+) pour chaque numéro d'ordre de la structure de données resultat. On cherche alors à définir la combinaison de chargement la plus pénalisante en terme de taux de restitution d'énergie :

$$\max_{q_i^- \leq q_i \leq q_i^+} G\left(\sum_i q_i Q_i\right) = \max_{i,j=1}^N \sum G_{ij} q_i q_j \quad \text{où } Q_i \text{ sont les } N \text{ chargements unitaires associés}$$

aux différents déplacements U_i contenus dans la structure de données resultat, et

$$G_{ij} = G(U_i, U_j) \text{ la forme bilinéaire de } G.$$

◆ NUME_ORDRE = num

Numéro d'ordre dans la structure de données resultat associé aux valeurs de contraintes bornes.

◆ VALE_MIN = qmin

Valeur minimal du coefficient appliqué au chargement associé au résultat stocké dans le numéro d'ordre num de la structure de données resu.

◆ VALE_MAX = qmax

Valeur maximale du coefficient appliqué au chargement associé au résultat stocké dans le numéro d'ordre num de la structure de données resu.

Remarques :

- *L'utilisateur doit donner autant de couples de bornes que de numéros d'ordre contenus dans la structure de données `resultat` sous peine d'erreur fatale.*
- *Cette option de calcul n'est valable que pour des calculs élastiques linéaires où la superposition de chargement par combinaison linéaire est possible.*
- *Un exemple d'utilisation de cette option pour maximiser G en présence de chargements signés et non signés est donné dans le 5.4.*

3.13 Mot-clé SIGNES

◇ BORNES =

Ce mot clé facteur est obligatoire si on utilise l'option '`CALC_K_MAX`'. Sinon il n'est pas utilisé. Il permet de définir, pour chaque numéro d'ordre de la structure de données `resultat`, le type de chargement, i.e. si le chargement est signé ou non signé.

◆ `CHARGE_S = listS`

Liste des numéros d'ordre de la structure de données `resultat` associés à des chargements signés.

◆ `CHARGE_NS = listNS`

Liste des numéros d'ordre de la structure de données `resultat` associés à des chargements non signés.

Cette option permet de distinguer ouverture et fermeture de la fissure :

$$K^{\max} = \sum_{CHARGE_S} K(u_i) + \sum_{CHARGE_NS} K(u_i).sign(K_I(u_i))$$

Un exemple d'utilisation de cette option est donné dans le 5.4.

3.14 Opérande TITRE◇ TITRE = titre
[U4.03.01].**3.15 Opérande INFO**◇ INFO = /1, [DEFAULT]
/2,

Niveau de messages dans le fichier '`MESSAGE`'.

3.16 Table produite

La commande *CALC_G* génère un concept de type *table*.
Celle-ci contient :

- en 2D ou 3D global : le taux de restitution d'énergie puis éventuellement, selon les options, sa dérivée ou les facteurs d'intensité des contraintes.
- en 3D local : pour l'option *CALC_G*, cette table contient, pour chaque nœud du fond de fissure :
 - le nom du nœud,
 - son abscisse curviligne le long du fond de fissure,
 - la valeur de *G* local au nœud.

Pour l'option *CALC_K_G*, la table contient :

- le numéro du point du fond de fissure (voir [U4.82.08]),
- son abscisse curviligne le long du fond de fissure,
- la valeur des facteurs d'intensité des contraintes K_1 , K_2 , K_3 locaux et du *G* local en chaque point ainsi que la valeur *BETA* de l'angle de propagation de la fissure.

L'angle *BETA* de propagation en 3D est calculé selon la formule suivante (cf. [R7.02.12]) :

$$BETA = 2 \arctan \left(\frac{1}{4} \left(\frac{K_I}{K_{II}} - \text{sign}(K_{II}) \sqrt{\left(\frac{K_I}{K_{II}} \right)^2 + 8} \right) \right)$$

En 2D, l'angle de propagation de la fissure est indiqué dans le fichier message si *INFO*=2 (cf. exemple dans le paragraphe 5.1).

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

4 Normalisation du taux de restitution global 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 $\mathcal{G}(\theta)$ issue de la commande *CALC_G* est indépendante du choix du champ θ :

$$G = \mathcal{G}(\theta) \quad , \quad \forall \theta \in \Theta$$

4.2 Axisymétrie

En axisymétrie il faut normaliser la valeur $\mathcal{G}(\theta)$ obtenue avec *Aster* :

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

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

4.3 3D

En dimension 3, la valeur de $\mathcal{G}(\theta)$ pour un champ θ donné est telle que :

$$\mathcal{G}(\theta) = \int_{\Gamma_o} G(s) \theta(s) \cdot \mathbf{m}(s) ds$$

Dans la commande *CALC_THETA* [U4.82.02], l'utilisateur définit la direction du champ θ en fond de fissure. Par défaut, c'est la normale au fond de fissure dans le plan des lèvres. En choisissant un champ θ unitaire au voisinage du fond de fissure, on a :

$$\theta(s) \cdot \mathbf{m}(s) = 1$$

et :

$$\mathcal{G}(\theta) = \int_{\Gamma_o} G(s) d\Gamma$$

Soit G le taux de restitution de l'énergie global, pour avoir la valeur de G par unité de longueur, il faut diviser la valeur obtenue par la longueur de la fissure l :

$$G = \frac{1}{l} \mathcal{G}(\theta) \quad \text{en 3D}$$

4.4 Symétrie du modèle

Ne pas oublier de multiplier par 2, les valeurs du taux de restitution d'énergie G ou $G(s)$ si on ne modélise que la moitié du solide par rapport à la fissure (ou préciser le mot clé *SYME_CHAR* = 'SYME' ou 'ANTI' dans les commandes concernées).

5 Exemples

5.1 Exemple d'utilisation en 2D

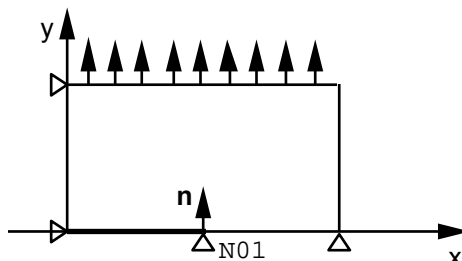


Figure 5.1-a : Calcul des facteurs d'intensités de contraintes.

```

ma = LIRE_MALLAGE ( )
mo = AFFE_MODELE ( MALLAGE = ma,
                   AFFE = _F( TOUT = 'OUI', PHENOMENE = 'MECANIQUE',
                               MODELISATION = 'D_PLAN' ) )

ff = DEFI_FOND_FISS ( NOEUD      = 'N01',
                     NORMALE     = (0. 1.), )

theta = CALC_THETA ( MODELE = mo,
                     THETA_2D = _F( NOEUD      = 'N01',
                                     MODULE     = 1.,
                                     R_INF      = 2.0,
                                     R_SUP      = 3.0,
                                     DIRECTION  = (1. 0.), ) )

G0 = CALC_G ( RESULTAT = resu,
              THETA     = _F( THETA = theta,
                              SYME_CHAR = 'SYME',
                              OPTION   = 'CALC _G'
                              INFO = 2, ) )

GK0 = CALC_G ( RESULTAT = resu,
              THETA     = _F( THETA = theta,
                              FOND_FISS = ff, )
              SYME_CHAR = 'SYME',
              OPTION   = 'CALC_K_G'
              INFO = 2, )

IMPR_TABLE ( TABLE = G4 )

```

On calcule les facteurs d'intensité de contraintes K_1 et K_2 sur le modèle `mo`, avec le déplacement `depl` solution du problème élastique avec :

- le champ de matériau `chma` produit par `AFFE_MATERIAU`,
- la charge `ch` produite par la commande `AFFE_CHAR_MECA`.

On récupère le nœud de fond de fissure `N01` et la normale à la fissure par le concept `fond_fiss`. On précise que le chargement global est symétrique par rapport à la fissure grâce au mot clé `SYME_CHAR`.

La commande `IMPR_TABLE` permet d'imprimer dans le fichier `RESULTAT` les différentes grandeurs calculées avec l'option `CALC_K_G`, à savoir le taux de restitution d'énergie `G`, les facteurs d'intensité des contraintes K_1 et K_2 , et `G_IRWIN`, taux de restitution de l'énergie obtenu à partir de la formule d'Irwin :

$$G_IRWIN = \frac{1}{E} (K_1^2 + K_2^2) \quad \text{en contraintes planes}$$

$$G_IRWIN = \frac{1-\nu^2}{E} (K_1^2 + K_2^2) \quad \text{en déformations planes}$$

avec E module de Young et ν coefficient de Poisson. La comparaison entre G et G_IRWIN permet de s'assurer de la cohérence des résultats.

Comme INFO vaut 2, l'angle de propagation de la fissure est également calculé, et le résultat est imprimé dans le fichier MESSAGE avec le format suivant :

Nœud de fond de fissure : N01

Coordonnées du nœud de fond de fissure : 0 . 0 .

Coordonnées de la normale à la fissure : 0 . 1 .

K_1	K_2	G (IRWIN)
2.14364E+01	0.0000E+00	1.14880E-03

Taux de restitution d'énergie G : 1.14907E-03

Direction de la déviation de la fissure (en degrés) :

Selon le critère K_1 maximum	: 0	avec K1 max	: 2.14364E+01
Selon le critère K_2 nul	: 0	avec K2 nul	: 0.0000E+00
Selon le critère G maximum	: 0	avec Gmax	: 1.1488E-03

A partir des facteurs d'intensité de contraintes K_1 et K_2 , on peut en effet calculer les coefficients K_1^* et K_2^* correspondant à une propagation de fissure donnée (d'après les travaux d'AMESTOY, BUI et DANG-VAN [R7.02.05 §2.5]).

La direction de la déviation de la fissure est calculée d'après ces résultats et selon 3 critères K_1^* maximum, K_2^* nul et G^* maximum. L'angle de propagation, donné en degré, est calculé par rapport au prolongement de la fissure.

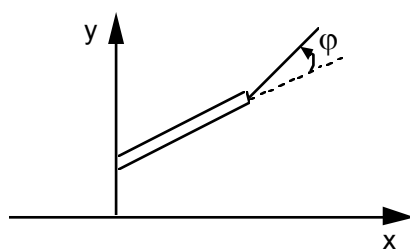


Figure 5.1-b : Angle de propagation

Remarques :

- Pour un chargement thermique, les coefficients caractéristiques du matériau (E , ν , ...) doivent être indépendants de la température.
- Attention à l'orientation de la normale à la fissure.

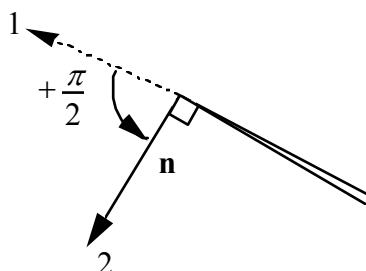


Figure 5.1-c : Orientation de la normale à la fissure

Pour d'autres exemples on pourra se reporter aux tests :

HPLP100 [V7.02.100] Plaque fissurée en thermoélasticité
SSLP103 [V3.02.103] Plaque circulaire fissurée

5.2 Exemple d'utilisation en 3D

Pour le calcul du taux de restitution de l'énergie en 3D (local ou global), le fond de fissure est défini dans DEFI_FOND_FISS :

```
ff=DEFI_FOND_FISS(  MAILLAGE=MA,
                    FOND_FISS=_F(GROUP_MA = 'LFF'),
                    NORMALE=(0., 0., 1.),
                    DTAN_ORIG=(1., 0., 0.),
                    DTAN_EXTR=(0., 1., 0.)),

G1LOC  =  CALC_G (  OPTION      =  'CALC_G',
                   RESULTAT    =  resu,
                   THETA       =  _F( FOND_FISS      =  ff,
                                     R_INF          =  1.,
                                     R_SUP          =  2.),
                   EXCIT       =  _F( CHARGE =  ch, ),
                   LISSAGE     =  _F( DEGRE  =  4.,
                                     LISSAGE_G = 'LAGRANGE',
                                     LISSAGE_THETA = 'LAGRANGE' ), )

G1GLOB =  CALC_G (  OPTION      =  'CALC_G_GLOB',
                   RESULTAT    =  resu,
                   THETA       =  _F( FOND_FISS      =  ff,
                                     R_INF          =  1.,
                                     R_SUP          =  2.),
                   EXCIT       =  _F( CHARGE =  ch, ), )
```

Pour le calcul des facteurs d'intensité des contraintes en 3D, le fond de fissure doit être défini par DEFI_FISS_XFEM :

```
ffx= DEFI_FISS_XFEM(MODELE=mo,
                   DEFI_FISS=_F( GROUP_MA_FISS='LEVREINF',
                                GROUP_MA_FOND='LFF'),
                   GROUP_MA_ENRI='VOLU',
                   ORIE_FOND=_F(PFON_INI=(0.1,0, 0.05 ),
                                VECT_ORIE=(0.0, 0.0, 1.0),
                                PT_ORIGIN=(0.0, 0.0, 0.0)),),),

KLOC  =  CALC_G (  OPTION      =  'CALC_K_G',
                   RESULTAT    =  resu,
                   THETA       =  _F( FISSURE      =  ffx,
                                     R_INF          =  1.,
                                     R_SUP          =  2.),
                   EXCIT       =  _F( CHARGE =  ch, ),
                   LISSAGE     =  _F( DEGRE  =  4.,
                                     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

5.3 Exemple de calcul de la dérivée du taux de restitution d'énergie pour une variation de domaine

Conservons la même configuration qu'au paragraphe précédent et calculons cette fois la dérivée de $G(\theta_f)$ par rapport à la variation de domaine pilotée par θ_s .

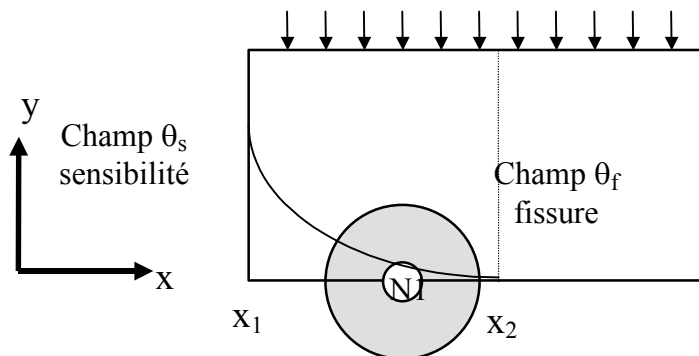


Figure 5.3-a : Dérivée de $G(\theta_f)$ par rapport à une variation de domaine pilotée par θ_s

Après avoir construit les modèles `mo` et `moth` en modélisation 'D_PLAN' et le champ θ_s (thetas), via le mot-clé `THETA_BANDE` de `CALC_THETA`, on affecte des chargements thermiques de type températures imposées sur les bords droit et gauche de la structure. Ensuite, on effectue le calcul thermique proprement dit qui utilise `thetas` (fourni via le mot-clé `SENSIBILITE`) pour calculer le champ de température et sa dérivée lagrangienne.

Remarques :

- Le calcul de sensibilité en thermique est restreint au cas linéaire 2D, stationnaire ou transitoire, avec des sources volumiques et des conditions de température imposée, de flux normal imposé et d'échange convectif. Les conditions d'échange entre paroi et de rayonnement ne sont pas encore prises en compte [R4.03.01] [U4.54.01].
- Les éléments finis supportant le maillage doivent être quadratiques.

```
mo = AFFE_MODELE ( MAILLAGE = ma,
                   AFFE = _F( TOUT = 'OUI', PHENOMENE = 'MECANIQUE',
                              MODELISATION = 'D_PLAN' ), )

moth = AFFE_MODELE ( MAILLAGE = ma,
                    AFFE = _F( TOUT = 'OUI', PHENOMENE = 'THERMIQUE',
                               MODELISATION = 'PLAN' ), )

thetas = CALC_THETA ( MODELE = mo,
                     OPTION = 'BANDE',
                     THETA_BANDE = _F( MODULE = 1.,
                                         R_INF = X1,
                                         R_SUP = X2 ), )

chther = AFFE_CHAR_THER( MODELE = moth,
                       TEMP_IMPO = _F( GROUP_NO = 'bordd', TEMP = -100 )
                       TEMP_IMPO = _F( GROUP_NO = 'bordg', TEMP = 100 ) )
```


Titre : Opérateur CALC_G
Auteur(s) : E. GALENNE

Date : 10/04/07
Clé : U4.82.03-H Page : 25/28

```
resth= THER_LINEAIRE(  MODELE =  moth,
                      SENSIBILITE = thetas,
                      CHAM_MATER = cmth,
                      EXCIT = (CHARGE = chther),)
```

Avant d'effectuer le calcul thermo-mécanique, on affecte une pression répartie sur le bord supérieur. Ce chargement, tout comme la température calculée, est utilisé dans la totalité du processus de sensibilité, contrairement aux chargements de pesanteur et aux déformations initiales ultérieures. Ces derniers n'interviendront qu'en post-traitement du calcul de mécanique.

Attention :

- Avec une telle prise en compte des chargements, l'utilisateur est seul responsable de l'interprétation des résultats.
- Pour être pleinement utilisable, le calcul de la sensibilité de G devra être étendu aux principaux chargements pour tout le processus.
- Le calcul de sensibilité en mécanique est restreint, pour l'instant, au cas linéaire D_PLAN ou $AXIS$ avec des conditions limites de types déplacement imposé, liaisons uniformes et pression externe [R4.03.01] [U4.51.01].

```
chmeca=  AFFE_CHAR_MECA(  MODELE =  mo,
                        TEMP_CALCULEE = resth,
                        DDL_IMPO = _F (GROUP_MA='bordi',DY = 0),
                        PRES_REP = _F (GROUP_MA='bords',PRES =100),)

resme= MECA_STATIQUE(  SENSIBILITE = thetas,
                      EXCIT = _F (CHARGE = chmeca))

a1 = 1.E-8
a2 = 2.E-6
a3 = 5.E-2
FONC1 = FORMULE(NOM_PARA = ('X','Y'), VALE = 'a1*(X**2)+a2*Y+a3')
FONC2 = FORMULE(NOM_PARA = ('X','Y'), VALE = 'a1*(Y**2)+a2*X+a3')
FONC3 = FORMULE(NOM_PARA = ('X','Y'), VALE = 'a1*(X*Y)+a3')

f=  AFFE_CHAR_MECA_F(  MODELE =  mo,
                    PESANTEUR = (10. 0. -1. 0.),
                    EPSI_INIT = _F ( TOUT = 'OUI',
                                    EPXX = fonc1,
                                    EPYY = fonc2,
                                    EPXY = fonc3))
```

Comme dans les exemples précédents on construit le champ θ_f (thetaf) qui va jouer le rôle d'une fonction test lors du calcul de l'intégrale du taux de restitution d'énergie $G(\theta_f)$ en délimitant la zone de calcul centrée autour du point N1.

```
thetaf =  CALC_THETA      (  MODELE =  mo,
                          THETA_2D =_F( NOEUD      =  'N1',
                                          MODULE      =  1.,
                                          R_INF        =  r1,
                                          R_SUP        =  r2,
                                          DIRECTION    =  (1. 0. 0.),),

G5 =  CALC_G (  RESULTAT      =  resme,
               SENSIBILITE    =  thetas,
               OPTION         =  'CALC_G',
               THETA          =  _F(THETA      =  thetaf,),
               EXCIT          =  (_F(CHARGE    =  chmeca),
                                _F(CHARGE    =  f)),
               SYME_CHAR      =  'SYME',
               COMP_ELAS      =  _F(RELATION = 'ELAS',
                                DEFORMATION = 'PETIT'),
               )
```

Pour d'autres exemples en D_PLAN on pourra se reporter au cas-test HPLP100B [V7.02.100]. On y trouvera notamment des exemples d'enchaînements de CREA_CHAMP permettant de construire des champs de contraintes analytiques et de translater un maillage (afin de simuler une différence finie en variation de domaine).

5.4 Maximisation de G et de K en présence de contraintes non signées

Cet exemple a pour but de préciser comment maximiser le taux de restitution de l'énergie pour un problème linéaire avec à la fois des contraintes signées (poids propre, pression interne) et des contraintes dont on ne connaît pas le signe a priori (séisme). Le problème étudié est en modélisation 3D, avec un comportement élastique linéaire. Deux types d'options sont utilisables :

- soit les options `G_MAX` / `G_MAX_GLOB` (maximisation du taux de restitution d'énergie) ;
- soit l'option `CALC_K_MAX` qui, basée sur le calcul de `K1`, permet de distinguer ouverture et fermeture de la fissure.

Dans les deux cas, le contact sur les lèvres de la fissure n'est pas pris en compte.

Supposons par exemple, qu'en plus des conditions aux limites de blocage `CHCL`, il y a un chargement de pression signé `CHPRES`, et deux chargements non signés s'appliquant sur des groupes de mailles distincts du modèle, `CH_NS1` et `CH_NS2` :

```
CHCL=AFFE_CHAR_MECA( MODELE=MO,
                      DDL_IMPO=( _F( GROUP_NO = 'SSUP_S', DZ = 0.),
                                   _F( GROUP_NO = 'SLAT_S', DX = 0.),
                                   _F( GROUP_NO = 'SAV_S', DY = 0.)),)

CHPRES=AFFE_CHAR_MECA( MODELE=MO,
                        PRES_REP=_F( GROUP_MA = 'SINF', PRES = -1.E6),)

CH_NS1=AFFE_CHAR_MECA( MODELE=MO,
                        FORCE_NODALE=_F( GROUP_NO = 'SLAT', FZ = 1540),)

CH_NS2=AFFE_CHAR_MECA( MODELE=MO,
                        FORCE_NODALE=_F( GROUP_NO = 'SINF', FX = 2100),)
```

On calcule la solution du problème associée à chacun de ces chargements en définissant des fonctions multiplicatrices :

```
F0=DEFI_FONCTION( NOM_PARA='INST',
                  PROL_GAUCHE='LINEAIRE',
                  PROL_DROITE='CONSTANT',
                  VALE=( 1., 1., 2., 0., 3., 0., ),)

F1=DEFI_FONCTION( NOM_PARA='INST',
                  PROL_GAUCHE='LINEAIRE',
                  PROL_DROITE='CONSTANT',
                  VALE=( 1., 0., 2., 1., 3., 0., ),)

F2=DEFI_FONCTION( NOM_PARA='INST',
                  PROL_GAUCHE='LINEAIRE',
                  PROL_DROITE='CONSTANT',
                  VALE=( 1., 0., 2., 0., 3., 1., ),)

LIST=DEFI_LIST_REEL( DEBUT=0.E+0,
                     INTERVALLE=_F( JUSQU_A = 3., NOMBRE = 3),)

RESU=MECA_STATIQUE( MODELE=MO,
                    CHAM_MATER=CHMAT,
                    EXCIT=( _F( CHARGE = CHCL),
                             _F( CHARGE = CHPRES, FONC_MULT = F0),
                             _F( CHARGE = CH_NS1, FONC_MULT = F1),
                             _F( CHARGE = CH_NS2, FONC_MULT = F2)),)
LIST_INST = LIST,
```

On définit le fond de fissure et la couronne thêta pour le calcul de G :

```
FOND=DEFI_FOND_FISS( MAILLAGE=MA,
                     FOND_FISS=_F( GROUP_MA = 'LFF'),
                     NORMALE=(0., 0., 1.),
                     DTAN_ORIG=(1., 0., 0.),
```

Titre : Opérateur CALC_G
Auteur(s) : E. GALENNE

Date : 10/04/07
Clé : U4.82.03-H Page : 27/28

```
DTAN_EXTR=(0., 1., 0.),)
```

```
THETA=CALC_THETA( MODELE=MO,  
                  FOND_FISS=FOND,  
                  THETA_3D=_F( TOUT = 'OUI',  
                                MODULE = 1.,  
                                R_INF = 0.2,  
                                R_SUP = 0.5),)
```

La maximisation de G se fait par l'option G_MAX_GLOB (en 3D global) ou par l'option G_MAX (en 3D local) de CALC_G. Le coefficient du chargement signé vaut 1, les coefficients des chargements non signés varient entre -1 et 1 :

```
G_MAX_G =  CALC_G(    THETA       =  _F(THETA,),  
                      RESULTAT   =  RESU,  
                      BORNES =   (_F( NUME_ORDRE = 1,  
                                      VALE_MIN = 1., VALE_MAX = 1.),  
                              _F( NUME_ORDRE = 2,  
                                      VALE_MIN = -1., VALE_MAX = 1.),  
                              _F( NUME_ORDRE = 3,  
                                      VALE_MIN = -1., VALE_MAX = 1.)),  
                      OPTION='G_MAX_GLOB',)  
  
IMPR_TABLE(TABLE = G_MAX_G)  
  
G_MAX_L =  CALC_G(    RESULTAT=RESU,  
                      THETA =  _F( FOND_FISS=FOND,  
                                      R_INF = 0.2,  
                                      R_SUP = 0.5,  
                                      MODULE = 1.0),)  
                      BORNES=( _F( NUME_ORDRE = 1,  
                                      VALE_MIN = 1., VALE_MAX = 1.),  
                              _F( NUME_ORDRE = 2,  
                                      VALE_MIN = -1., VALE_MAX = 1.),  
                              _F( NUME_ORDRE = 3,  
                                      VALE_MIN = -1., VALE_MAX = 1.)),  
                      OPTION='G_MAX',)  
  
IMPR_TABLE(TABLE = G_MAX_L)
```

La table produite par CALC_G option 'G_MAX_GLOB' est la suivante :

```
#ASTER 8.02.01 CONCEPT GMAX_G CALCULE LE 21/12/2005 A 15:49:17 DE TYPE  
#TABLE_SDASTER  
Q_1           Q_2           Q_3           G           G_MAX  
1.00000E+00   1.00000E+00   -1.00000E+00   3.91703E+03   3.91703E+03  
1.00000E+00   1.00000E+00   -1.00000E+00   3.91703E+03   3.91703E+03  
1.00000E+00   -1.00000E+00   -1.00000E+00   3.63507E+03   -  
1.00000E+00   -1.00000E+00   -1.00000E+00   3.63507E+03   -  
1.00000E+00   -1.00000E+00   1.00000E+00   2.92029E+03   -  
1.00000E+00   -1.00000E+00   1.00000E+00   2.92029E+03   -  
1.00000E+00   1.00000E+00   1.00000E+00   2.68007E+03   -  
1.00000E+00   1.00000E+00   1.00000E+00   2.68007E+03   -
```

Ainsi, le taux de restitution maximum est obtenu pour la combinaison du chargement de pression avec CH_NS1 avec un signe '+' et CH_NS2 avec un signe '-'. Les options G_MAX / G_MAX_GLOB ne permettent pas de distinguer les chargements tendant à ouvrir la fissure de ceux qui tendent à la fermer : il peut donc arriver que le maximum de G trouvé corresponde à une refermeture de la fissure.

Pour utiliser l'option CALC_K_MAX, il faut préalablement redéfinir le fond de fissure avec DEFI_FISS_XFEM :

```
FFX=DEFI_FISS_XFEM(MODELE=MO,
```

Titre : *Opérateur CALC_G*
Auteur(s) : **E. GALENNE**

Date : 10/04/07
Clé : U4.82.03-H Page : 28/28

```

DEFI_FISS=_F(GROUP_MA_FISS='SSUP_S',
              GROUP_MA_FOND='LFF',),
GROUP_MA_ENRI='VTOT',
ORIE_FOND=_F(PFON_INI=(-0.005,0.0,0.0,)),
              VECT_ORIE=(0.0,1.0,0.0,)),
              PT_ORIGIN=(0.0,0.0,0.0,)),)
);

K_MAX = CALC_G ( RESULTAT=RESU,
                 THETA = _F( FISSURE=FFX,
                             R_INF =0.2,
                             R_SUP =0.5,
                             MODULE=1.0,)),
                 SIGNES= F( CHARGE_S = (1,)),
                           CHARGE_NS = (2,3),),
                 OPTION='CALC_K_MAX',)

IMPR_TABLE(TABLE = K_MAX)

```

Dans le tableau résultat produit par cette option, on trouve successivement :

- les facteurs d'intensité des contraintes K1, K2 et K3 et le G associés à chacun des chargements (fonctions de l'abscisse curviligne) ;
- les facteurs d'intensité des contraintes maximum, le G maximum et les coefficients des charges associés (1 pour les chargements signés, +/-1 pour les chargements non signés).

#K_MAX AVEC R_INF = 0.2 ET R_SUP = 0.5									
Q_1	Q_2	Q_3	NUM_PT	ABS_CURV	K1_LOCAL	K2_LOCAL	K3_LOCAL	G_LOCAL	
1	0	0	1	0.00000E+00	8.46799E+05	-3.39509E+05	1.65142E+03	5.85407E+00	
1	0	0	2	3.92069E-01	8.51341E+05	-3.48645E+05	2.89982E+03	5.85097E+00	
...									
0	1	0	1	0.00000E+00	5.20948E+05	-2.31674E+05	-2.01759E+04	2.19826E+00	
0	1	0	2	3.92069E-01	5.25149E+05	-2.42255E+05	-7.40016E+03	2.19755E+00	
...									
0	0	1	0	0.00000E+00	-1.97960E+05	8.03611E+04	-1.61407E+04	1.52288E+01	
0	0	1	0	3.92069E-01	-1.99557E+05	9.20569E+04	-5.92013E+03	1.52227E+01	
...									
1	1	-1	1	0.00000E+00	1.56571E+06	-6.59219E+05	-2.38376E+03	7.64144E+01	
1	1	-1	2	3.92069E-01	1.57605E+06	-6.82957E+05	1.41979E+03	7.63789E+01	
...									

Pour d'autres exemples, on peut se reporter au cas test SSLV134E/F [V3.04.134].

PAGE LAISSEE BLANCHE INTENTIONNELLEMENT