Opérateur PROPA_FISS

1 But

Le but de l'opérateur PROPA_FISS est de faire propager une fissure, cette fissure étant définie avec la méthode X-FEM. Trois types de méthodes sont disponibles :

- une méthode basée sur un algorithme de *fast marching* (METHODE_PROPA='SIMPLEXE', 'UPWIND') : cette méthode est disponible en 2D et 3D pour tous types d'éléments finis. La fissure initiale peut être définie sur le maillage du modèle ou sur un maillage auxiliaire.
- une méthode purement géométrique par calcul de distances au front de fissure (METHODE_PROPA='GEOMETRIQUE')
- une méthode utilisant un maillage intermédiaire surfacique de la lèvre de la fissure (METHODE_PROPA='MAILLAGE'): cette méthode est disponible en 2D et en 3D pour tous types d'éléments finis. La fissure initiale doit elle aussi être définie à l'aide d'un maillage.

L'opérateur PROPA_FISS calcule ainsi la fissure résultante d'une propagation en fatigue d'après une loi de Paris locale, à partir de la fissure précédente, et du tableau des facteurs d'intensité des contraintes issus de CALC_G ou de POST_K1_K2_K3.

Pour les fissures en demi-droite (en 2D) ou planes à fond rectiligne (en 3D), le maillage surfacique de la fissure initiale peut être créé par l'opérateur PROPA_FISS (METHODE_PROPA = 'INITIALISATION').

L'opérateur produit :

- soit un concept de type fiss_xfem définissant la fissure propagée (cas METHODE_PROPA='SIMPLEXE', 'UPWIND' ou 'GEOMETRIQUE') ;
- soit le maillage de la fissure initiale (cas METHODE_PROPA='INITIALISATION') ;
- soit le maillage de la fissure propagée (cas METHODE_PROPA='MAILLAGE') .

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

L'aspect théorique de la propagation des level-sets pour la méthode X-FEM est abordé dans [R7.02.12].

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

2 Syntaxe

```
PROPA_FISS(

      ♦ METHODE_PROPA   =  / 'SIMPLEXE'
                           / 'UPWIND'
                           / 'MAILLAGE'
                           / 'INITIALISATION'
                           / 'GEOMETRIQUE'


#   Si METHODE_PROPA = 'GEOMETRIQUE'
      ◊ OPERATION    = / 'RIEN'                              [DEFAUT]
                       / 'DETEC_COHESIF'
                       / 'PROPA_COHESIF'

#   Si METHODE_PROPA = 'SIMPLEXE' ou 'UPWIND' ou 'GEOMETRIQUE'
      ♦   MODELE       = mo,                                 [modele]
      ◊   TEST_MAIL    = / 'NON'                             [DEFAUT]
                        / 'OUI'
      # Si TEST_MAIL='OUI'
          ◊ ITERATIONS   =  / iters,                         [I]
                            / 5,                             [DEFAUT]
          ◊ TOLERANCE =  / toler,                            [R]
                         / 5.0,                              [DEFAUT]
      # Fin si

      ◊   ZONE_MAJ     = / 'TORE'                            [DEFAUT]
                         / 'TOUT'

      # Si ZONE_MAJ='TORE'
          ◊ RAYON_TORE   = raytor,                           [R]

      ♦ FISSURE        = _F(
            ♦ FISS_ACTUELLE  = fiss,                         [fiss_xfem]
            ♦ FISS_PROPAGEE  = CO('FISS'),                   [TXM]

            # Si TEST_MAIL='NON',
               ♦ TABLE          = sif ,                      [table]
               ◊ NB_POINT_FOND  = nbptfd,                    [I]
            # Fin si
                        ),
       # Si OPERATION='RIEN' ou OPERATION='PROPA_COHESIF'
      ♦ DA_MAX      = da,                                    [R]

       # Si OPERATION='DETEC_COHESIF'
      ♦ RESULAT     = res,                                   [evol_noli]

      # Si OPERATION='RIEN'
      ♦ RAYON       = ray ,                                  [R]

      # Si TEST_MAIL='NON'et OPERATION='RIEN',
      ♦ COMP_LINE   = F(   ♦   COEF_MULT_MINI = cmin,        [R]
                           ♦   COEF_MULT_MAXI = cmax,        [R]
                     ),

#   Si METHODE_PROPA = 'MAILLAGE'
       ♦ MAIL_STRUC    =  mast,                              [maillage]
```


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

```
        ♦ ITERATION    =  iter,                              [ I ]
        ♦ DA_MAX       =  da,                                [R]

        ♦  FISSURE     =   _F(
             ♦  MAIL_ACTUEL   = max1                         [ maillage ]
             ♦  FISS_ACTUELLE=  FISS1                        [ fiss_xfem]
             ◊  MAIL_PROPAGE =  CO('max2'),                  [TXM]
             ◊  GROUP_MA_FOND=  /  'FOND'                    [DEFAUT]
                                /  grmaff                    [TXM]
             ◊  GROUP_MA_FISS=  /  'FISS'                    [DEFAUT]
                                /  grmafi,                   [TXM]
             ♦  TABLE        =  sif,                         [table]


      ♦ MAIL_TOTAL =  CO('matot2'),                        [TXM]

      ♦ COMP_LINE  = _F( ♦  COEF_MULT_MINI = cmin,          [R]
                         ♦  COEF_MULT_MAXI = cmax,          [R]
                       ),
                         ),

#  Si METHODE_PROPA ='INITIALISATION',
      ♦  MAIL_STRUC   =  mast,                              [maillage]
      ♦  FORM_FISS    =  /  'DEMI_DROITE'
                         /  'DEMI_PLAN'
                         /  'ELLIPSE'
      ◊  GROUP_MA_FOND=  /  'FOND'                          [DEFAUT]
                         /  grmaff                          [TXM]
      ◊  GROUP_MA_FISS=  /  'FISS'                          [DEFAUT]
                         /  grmafi,                         [TXM]
      ◊  MAIL_FISS    =  CO('max2'),                        [TXM]
      ♦  MAIL_TOTAL   =  CO('matot2'),                      [TXM]

   #  Cas DEMI_DROITE (en 2D)
         ♦  PFON   =  (xf,yf,zf),                           [l_R]
         ♦  DTAN   =  (xt,yt,zt),                           [l_R]
   #  Cas DEMI_PLAN (en 3D)
         ♦  POINT_ORIG   =  (xo,yo,zo),                     [l_R]
         ♦  POINT_EXTR   =  (xe,ye,ze),                     [l_R]
         ♦  DTAN         =  (xt,yt,zt),                     [l_R]
         ♦  NB_POINT_FOND=  nbfond,                         [I]
   #  Cas ELLIPSE (en 3D)
         ♦  CENTRE           =  (xo,yo,zo),                 [l_R]
         ♦  DEMI_GRAND_AXE   =  a ,                         [R]
         ♦  DEMI_PETIT_AXE   =  b,                          [R]
         ♦  VECT_X           =  (vxx, vxy, vxz),            [l_R]
         ♦  VECT_Y           =  (vyx, vyy, vyz),            [l_R]
         ♦  ANGLE_ORIG       =  ango,                       [R]
         ♦  ANGLE_EXTR       =  ange,                       [R]
         ♦  NB_POINT_FOND    =  nbpf,                       [I]

#  Fin si

# Si (METHODE_PROPA ='SIMPLEXE' ou 'UPWIND' et TEST_MAIL='NON')
# ou ( METHODE_PROPA ='MAILLAGE')
♦  LOI_PROPA =  _F (
      ♦  LOI    =  /'PARIS',
      # Si LOI='PARIS'
         ♦   M      = m,                                   [R]
         ♦   C      = c,                                   [R]
```


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

```
            ◊  DELTA_K_SEUIL=  /  0.                          [DEFAUT]
                               /   val_seuil ,                [R]
         # Fin si

         ♦   MATER      = ma ,                                 [mater]
                       ),
      ◊  CRIT_ANGL_BIFURCATION  =  /  'SITT_MAX'               [DEFAUT]
                                   /  'SITT_MAX_DEVER'
                                   /  'K1_MAX'
                                   /  'K2_NUL'
                                   /  'PLAN'
                                   /  'ANGLE_IMPO'
                                   /  'ANGLE_IMPO_GAMMA'
                                   /  'ANGLE_IMPO_BETA_GAMMA'

      #  Impression d'informations
         ◊  INFO   =  /  0
                      /  1                                     [DEFAUT]
                      /  2
      )
```


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

3 Opérandes communs à toutes les options

3.1 Opérande METHODE_PROPA

♦ METHODE_PROPA = / 'SIMPLEXE',
/ 'UPWIND',
/ 'MAILLAGE',
/ 'INITIALISATION',
/ 'GEOMETRIQUE',

Trois méthodes sont disponibles pour faire propager une fissure définie avec la méthode X-FEM :

•Méthodes basées sur un algorithme de fast marching

(METHODE_PROPA='UPWIND') : une discrétisation de l'équation Eikonale est résolue pour réinitialiser lsn et lst. L'utilisation de la grille auxiliaire est indispensable pour les maillages irréguliers. Pour plus de détails concernant les algorithmes relatifs à cette méthode, on pourra se référer à la documentation de référence [R7.02.12].

(METHODE_PROPA='SIMPLEXE') : Méthode robuste, qui peut être appliquée à des maillages quelconques. Pour plus de détails concernant les algorithmes relatifs à cette méthode, on pourra se référer à la documentation de référence [R7.02.12].

•une méthode utilisant un maillage intermédiaire surfacique de la lèvre de la fissure (METHODE_PROPA='MAILLAGE') : cette méthode est disponible en 2D et en 3D pour tous types d'éléments finis.
La fissure initiale doit elle aussi être définie à l'aide d'un maillage, avec des règles de nommage spécifiques. Par exemple en 3D, le fond s'appelle FOND_0 et contient les nœuds NXA1 , NXB1 , … ; la lèvre s'appelle FISS_0 et contient les mailles surfaciques MXA1 , MXB1 , … ; la maille surfacique MXA1 contient les nœuds NXA0 , NXB0 , NXA1 et NXB1 .

•une méthode utilisant une approche purement géométrique (METHODE_PROPA='GEOMETRIQUE'): cette méthode utilisant une propagation vectorielle du fond de fissure et de sa base locale ainsi qu'une estimation des distances par rapport à ce dernier dans cette base locale est disponible en 2D et en 3D pour tous types d'éléments finis sans aucune restriction.

Pour faciliter la définition de cette fissure initiale dans les cas les plus usuels (fissure en demi-droite (en 2D) ou plane à fond rectiligne (en 3D)), il faut utiliser la méthode 'INITIALISATION'.

Remarque sur les différentes méthodes :

La méthode de projection sur un maillage intermédiaire est quant à elle plus approchée, mais elle permet d'avoir des résultats très satisfaisants notamment pour les propagations en mode I pur.

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

Les méthodes de fast marching sont robustes et donnent des résultats très satisfaisants pour les propagations en mode mixte. La méthode simplexe ne nécessite pas de grille auxiliaire.

La méthode purement géométrique est une méthode qui ne nécessite pas de grille auxiliaire et offre de bonnes performances.

3.2 Opérande OPERATION

```
#  Si METHODE_PROPA = 'GEOMETRIQUE'
      ◊ OPERATION   = / 'RIEN'                              [DEFAUT]
                      / 'DETEC_COHESIF'
                      / 'PROPA_COHESIF'
```

Ce mot-clé définit la nature de l'opération souhaitée lors de l'actualisation de la fissure :

Si 'RIEN' , on souhaite une propagation classique des level-sets normales et tangentes selon un certain angle et une certaine avancée. C'est l'opération qu'il faut sélectionner en l'absence de forces cohésives.

L'opération ' DETEC_COHESIF ' effectue une détection de front par post-traitement d'un résultat avec des forces cohésives. Etant donnée une surface de fissuration possible, sur laquelle sont définies des forces de cohésion, elle permet de détecter la nouvelle position du front de propagation, par post-traitement du résultat, qui doit être donné sous le mot-clé RESULTAT . Il s'agit donc d'une actualisation de level-set tangente uniquement.

L'opération ' PROPA_COHESIF ' étend la surface de fissuration possible selon un angle donné en argument dans TABLE , et d'une longueur uniforme spécifiée par DA_MAX . Il s'agit, une fois la direction de propagation déterminée, d'étendre la zone de fissuration possible afin de réaliser le prochain calcul mécanique. Cette extension doit se faire sur une longueur supérieure à l'avancée effective de la fissure. Ceci peut être quantifié par une analyse dimensionnelle simple, et doit être vérifié *a posteriori* au cours du calcul.

Les algorithmes de propagation de fissure avec éléments cohésifs sont décrits dans la documentation de référence [R7.02.19].

3.3 Mot clé LOI_PROPA

```
#  Si  ( METHODE_PROPA ='SIMPLEXE'  ou    'UPWIND'  ou  'GEOMETRIQUE'  et
RAFF_MAIL='NON')
# ou  ( METHODE_PROPA ='MAILLAGE' ou 'INITIALISATION')
♦  LOI_PROPA =  _F  (
   ♦  LOI    =  /'PARIS',
      # Si LOI='PARIS'
         ♦   M      = m,                                    [R]
         ♦   C      = c,                                    [R]
         ◊  DELTA_K_SEUIL=  /  0.                           [DEFAUT]
                            /   val_seuil ,                 [R]
      # Fin si
♦ MATER    = ma ,                                           [ mater ] ),
```

Le mot clé facteur LOI_PROPA définit la loi de propagation utilisée.

Pour l'instant, la seule loi disponible est une loi de Paris,dans la forme suivante:

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

$$\frac{da}{dN}=C.\left(\Delta K_{eq}-\Delta K_{seuil}\right)^{m}$$

Les coefficients C et m de cette loi doivent obligatoirement être renseignés. Le seuil de propagation ΔK_{seuil} est réglable via le paramètre optionnel DELTA_K_SEUIL, dont la valeur par défaut est de 0. Dans le cas $\Delta K_{eq}<\Delta K_{seuil}$, l'avancée est nulle.

Remarque :

Pour l'instant toutes les fissures du modèle doivent propager dans le même matériau. En fait un seul matériau peut être donné et il est utilisé pour toutes les fissures du modèle.
En plus la loi de Paris ne permet pas de considérer le seuil de propagation de la fissure.

Attention :

Les constantes de la loi de propagation doivent être données en façon telle que l'unité de la vitesse d'avancement da/dN est égale à $L/cycles$, où L est l'unité utilisée dans le modèle pour la longueur.

3.4 Opérande RESULTAT

```
# Si OPERATION='DETEC_COHESIF'
      ♦ RESULAT     = res,                                  [evol_noli]
```

Il s'agit de fournir le résultat de calcul de STAT_NON_LINE dont le champ de variables internes de la loi cohésive permettra de déterminer la position du nouveau front de propagation.

3.5 Opérande DA_MAX

```
♦ DA_MAX =  da,                                               [R]
```

Dans le cas où il y a une seule fissure dans le modèle mo, cet opérande définit l'avance maximale de la fissure.

Dans le cas où il y a plusieurs fissures dans le modèle mo, cet opérande définit l'avance maximale de la fissure qui propage plus vite. Le nombre de cycles de fatigue est calculé en utilisant la vitesse du point du fond de la fissure qui propage la plus vite. Ce nombre de cycles est utilisé pour toutes les fissures du modèle.

Si OPERATION='PROPA_COHESIF', ce mot-clé définit la longueur uniforme selon laquelle on étend la nouvelle surface de fissuration possible en vue du prochain pas de propagation.

3.6 Mot clé facteur COMP_LINE

```
 # Si TEST_MAIL='NON',
♦  COMP_LINE = _F( ♦ COEF_MULT_MINI = cmin,              [R]
                   ♦ COEF_MULT_MAXI = cmax,              [R]
 ),
```

Les paramètres COEF_MULT_MINI et COEF_MULT_MAXI de l'opérande COMP_LINE définissent les conditions de chargement minimal et maximal du cycle de fatigue. Les deux valeurs sont les constantes par lesquelles on doit multiplier les chargements de la configuration de référence pour obtenir les conditions de chargement minimal et maximal du cycle de fatigue. Donc si la configuration de référence choisie coïncide avec la condition de chargement minimal ou maximal, la valeur de COEF_MULT_MINI ou COEF_MULT_MAXI, respectivement, est égale à 1.

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

3.7 Mot clé facteur CRIT_ANGL_BIFURCATION

```
◊ CRIT_ANGL_BIFURCATION = /  'SITT_MAX '                [DEFAUT]
                         /  'SITT_MAX_DEVER'
                         /  'K1_MAX'
                         /  'K2_NUL'
                         /  'PLAN'
                         /  'ANGLE_IMPO'
                         /  'ANGLE_IMPO_GAMMA'
                         /  'ANGLE_IMPO_BETA_GAMMA'
```

Cet opérande permet de choisir le critère de calcul de l'angle de bifurcation :

- 'SITT_MAX' : critère de la contrainte circonférentielle maximale (Maximal Hoop Stress criterion, [R7.02.05 §2.5.2]) c'est le critère par défaut. Il est calculé à partir des valeurs de K1 et K2. Il est disponible en 2D et en 3D.
- ' SITT_MAX_DEVER ': critère de la contrainte circonférentielle maximale avec prise en compte du KIII [R7.02.05 §2.5.2] et prise en compte de la rotation de déversement [R7.02.13 §2.1]. Il est calculé à partir des valeurs de K1, K2 et K3 et disponible uniquement en 3D.
- 'K1_MAX' , 'K2_NUL' : critères d'Amestoy, Bui et Dang-Van [R7.02.05 §2.5.1]. Ces 2 critères sont uniquement disponibles en 2D. L'angle est donné à 10 degrés près. Attention, on constate que le critère 'K2_NUL' ne fonctionne pas pour un angle supérieur à 60°.
- 'PLAN' : définit un angle nul correspondant à une propagation plane.
- 'ANGL_IMPO' : permet de choisir un angle préalablement calculé. Dans ce cas, on prend comme angle celui donné dans la colonne BETA de la table en entrée (table donnée sous le mot-clé TABLE) et la rotation de déversement n'est pas activée. Il faut donc pour cela que la table en entrée de PROPA_FISS contienne une colonne BETA .
- 'ANGLE_IMPO_GAMMA': permet de choisir un angle de déversement préalablement calculé. Dans ce cas, on prend comme angle de déversement celui donné dans la colonne GAMMA de la table en entrée (table donnée sous le mot-clé TABLE) et l'angle de branchement est calculé [R7.02.13§2.1]. Il faut donc pour cela que la table en entrée de PROPA_FISS contienne une colonne GAMMA . L'angle de déversement BETA est calculé par l'opérateur POST_RUPTURE, dans la configuration CRITERE = SITT_MAX_DEVER [U4.82.04].
- 'ANGLE_IMPO_BETA_GAMMA': permet de choisir des angles de branchement et de déversement préalablement calculés. Dans ce cas on prend comme angle de branchement et de déversement respectivement BETA et GAMMA donnés dans la table en entrée (sous le mot-clé TABLE). Il faut pour cela que la table en entrée contienne une colonne BETA et une colonne GAMMA de même taille.

Remarque :

La possibilité d'imposer l'angle de branchement BETA puis d'en déduire l'angle de déversement GAMMA n'est pas conforme au modèle. En effet le calcul de GAMMA impose la connaissance des facteurs d'intensité des contraintes que l'on ne peut pas retrouver à partir de la seule donnée de l'angle de branchement BETA.

3.8 Opérande INFO

/ 0 : impression sur le fichier 'MESSAGE'

- des étapes de calcul

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

/ 1 : impression sur le fichier 'MESSAGE'

- des mêmes informations qu'en INFO=1
-
  - méthode 'MAILLAGE' et 'INITIALISATION' : impression du maillage surfacique de la fissure au format Aster

/ 2 : Impression sur le fichier 'MESSAGE'

- des même informations qu'en INFO=1
- méthode 'UPWIND', 'SIMPLEXE': des détails du calcul des champs et d'enrichissement de la SD fissure
- méthode 'UPWIND' : tableau de connexion des nœuds de la grille du calcul

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

4 Opérandes spécifiques aux méthodes SIMPLEXE, UPWIND et GEOMETRIQUE

4.1 Opérande MODELE

♦ MODELE = mo, [modele]

Nom du modèle sur lequel la fissure à propager est définie.

4.2 Opérande TEST_MAIL

◊ TEST_MAIL = / 'NON' [DEFAUT]
/ 'OUI'

Le maillage utilisé pour la représentation des level sets (maillage du modèle mo) doit être suffisamment raffiné pour bien représenter la fissure. On peut le vérifier en utilisant l'opérande TEST_MAIL.

Dans le cas TEST_MAIL='OUI', la même vitesse de propagation est affectée à tous les points des fonds de fissures définies sur le modèle mo. Toutes les fissures avancent donc de la longueur donnée par l'opérande DA_MAX. Si le maillage est suffisamment raffiné, le fond de fissure après la propagation doit être homothétique au fond avant la propagation. Si cela n'est pas vérifié, le maillage doit être raffiné.

L'opérande calcule plusieurs propagations des fissures du modèle mo (voir opérandes DA_MAX, ITERATIONS et TOLERANCE) et pour chaque propagation il calcule la distance entre le nouveau fond après la propagation et le fond au début de la propagation (voir FISS_ACTUELLE dans l'opérande FISSURE). Si le maillage du modèle mo ou la grille auxiliaire griaux utilisés pour la représentation des level sets sont suffisamment raffinés, la distance calculée doit être égale à l'avance imposée DA_MAX multipliée par le numéro de l'itération courante. Une petite tolérance est utilisée pour cette vérification (voir opérande TOLERANCE). Si cela n'est pas vérifié, un message d'alerte est issue par PROPA_FISS.

La résolution du modèle fissuré et le calcul des facteurs d'intensité des contraintes ne sont pas nécessaires quand on utilise cet opérande. La vérification est donc très rapide (voir cas test zzzz255a et zzzz255b).

4.3 Opérande ITERATIONS

```
# Si TEST_MAIL='OUI'
 ◊ ITERATIONS   =  / iters,                    [I]
                   / 5,                        [DEFAUT]
```

Cet opérande est utilisé pour donner le nombre d'itérations (propagations) calculées pendant le test du maillage (voir opérande TEST_MAIL). Pour chaque itération, la fissure est propagée et la distance entre le fond de fissure propagé et le fond au début de la propagation (voir FISS_ACTUELLE dans l'opérande FISSURE) est calculée et vérifiée. Il est fortement conseillé d'utiliser un nombre d'itérations égal ou supérieur à 5 car les erreurs numériques s'accumulent à chaque itération.

4.4 Opérande TOLERANCE

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

```
# Si TEST_MAIL='OUI'
◊ TOLERANCE   =   / toler,                                  [R]
                  / 5.0,                                    [DEFAUT]
```

Cet opérande est utilisé pour donner la tolérance, exprimée comme pourcentage, utilisée dans le test du maillage (voir opérande TEST_MAIL). Pour chaque itération du test (voir opérande ITERATIONS), la distance maximale et minimale entre le fond propagé et le fond au début de la propagation (voir FISS_ACTUELLE dans l'opérande FISSURE) sont calculées. Théoriquement ces distances sont identiques et leur valeur est égale à l'avance imposé DA_MAX multipliée par le numéro de l'itération courante. On calcule donc la différence entre les distances maximale et minimale et la distance théorique et on vérifie que les deux différences calculées sont inférieures à une valeur limite. Cette valeur limite est calculée comme pourcentage de la longueur de la plus petite arête des éléments du maillage utilisée pour la représentation des level sets. Le pourcentage utilisé est donné par l'opérande TOLERANCE.

4.5 Opérande ZONE_MAJ

```
◊ ZONE_MAJ        =   / 'TORE'                              [DEFAUT]
                      / 'TOUT'
```

Cet opérande permet de sélectionner la zone utilisée pour la mise à jour des level sets. On peut choisir de faire le calcul sur tous les nœuds du modèle mo en utilisant ZONE_MAJ='TOUT' ou de limiter la zone de calcul à les nœuds du modèle mo qui sont dans un tore construit autour du fond de la fissure en utilisant ZONE_MAJ='TORE' .
Ce dernier choix permet de réduire fortement le temps de calcul même si le maillage utilisé pour la représentation des level sets n'est pas formé par un grand nombre de mailles. Dans le cas de propagation en mode mixte, l'utilisation de la restriction de la zone de mise à jour est fortement conseillée pour améliorer la performance du calcul et pour éviter la distorsion des level sets, qui peux causer une erreur fatale pendant la mise à jour.
Le calcul du rayon du tore de localisation et sa mise à jour à chaque propagation sont fait de façon automatique. Toutefois, dans le cas où la taille des éléments dans la zone de propagation n'est pas uniforme, le rayon calculé peut être trop petit et un message d'erreur est issu par l'opérateur PROPA_FISS . Dans ce cas l'utilisateur est obligé de donner explicitement une valeur du rayon (voir opérande RAYON_TORE).

4.6 Opérande RAYON_TORE

```
 # Si ZONE_MAJ='TORE'
       ◊ RAYON_TORE = raytor,                               [R]
```

Cet opérande permet de donner la valeur du rayon du tore utilisé pour la restriction de la zone de mise à jour des level sets (voir opérande ZONE_MAJ). Le rayon du tore est calculé automatiquement par l'opérateur PROPA_FISS et l'utilisateur n'est pas obligé d'utiliser cet opérande. Toutefois la valeur calculée peut être trop petite dans certain cas, par exemple quand les tailles minimale et maximale des éléments dans la zone de propagation sont très différentes entre elles. Dans tous ces cas, l'utilisateur doit forcement donner le rayon du tore en suivant les conseilles contenus dans les messages d'erreur issus par PROPA_FISS.
Il faut remarquer que la valeur du rayon ne peut pas changer beaucoup entre deux propagations successives (voir la documentation R7.02.12 pour plus de détails). Donc il est fortement conseillé d'utiliser la même valeur à chaque appel de PROPA_FISS en donnant la valeur maximale prévue. La valeur du rayon doit être toujours supérieure à une valeur minimale qui peut être estimée en calculant l'addition des valeurs de l'avancée de la fissure (voir opérande

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

DA_MAX) et du rayon de convergence (voir opérande RAYON). Toutefois cette valeur est liée aussi à la taille des éléments dans le tore et donc il est fortement conseillé de ne pas utiliser une valeur inférieure à celle qui a été calculée automatiquement par PROPA_FISS . Une erreur est issue par PROPA_FISS si le rayon donné n'est pas bonne par rapport à la propagation à simuler et à la taille des éléments du maillage.

Attention : dans les cas de fissures avec plusieurs fonds de fissure (par exemple dans le cas d'une fissure de type segment, en 2d, qui possède 2 fonds de fissure) ; il ne faut pas que le tore englobe les autres fonds de la fissure (même si ces autres fonds se trouvent « en dehors du maillage »).

4.7 Mot clé facteur FISSURE

Cet opérande obligatoire permet d'indiquer les fissures du modèle mo qui propagent.

4.7.1 Syntaxe FISSURE

```
♦  FISSURE      = _F(
     ♦ FISS_ACTUELLE  = fiss,                          [fiss_xfem]
     ♦ FISS_PROPAGEE  = CO('FISS'),                    [TXM]

     # Si RAFF_MAIL   = 'NON',
        ♦    TABLE           = sif ,                   [table]
        ◊    NB_POINT_FOND   = nbptfd,                 [I]
     # Fin si
                  ),
```

4.7.2 Opérande FISS_ACTUELLE

```
♦  FISS_ACTUELLE   = fiss,                             [ fiss_xfem ]
```

On donne le nom de la fissure actuelle à propager. Cette fissure doit être déjà affectée au modèle mo en utilisant l'opérateur MODI_MODELE_XFEM.

4.7.3 Opérande FISS_PROPAGEE

```
♦  FISS_PROPAGEE   = CO('FISS'),                       [ TXM ]
```

Cet opérande permet de définir le nom de la fissure propagée produit par l'opérateur PROPA_FISS. Le résultat produit est un concept de type fiss_xfem définissant la fissure propagée.

4.7.4 Opérande TABLE

```
 #  Si TEST_MAIL='NON'
◊  TABLE  =  sif,                                      [table]
```

Cet opérande doit être renseigné avec le tableau des facteurs d'intensité des contraintes en fond de la fissure donnée par l'opérande FISS_ACTUELLE. Ce tableau provient des opérateurs CALC_G. ou POST_K1_K2_K3.
Dans le cas de fond de fissure multiple (voir R7.02.12, paragraphe «Fond de fissure multiple»), pour lequel le fond de la fissure est formé par plusieurs morceaux, l'opérateur CALC_G/POST_K1_K2_K3 doit être appelé plusieurs fois, une fois pour chaque morceau (opérande NUME_FOND de CALC_G/POST_K1_K2_K3). On a donc plusieurs tableaux, un

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

tableau pour chaque morceau, qui doivent être combinés dans un seul tableau qui sera donné à PROPA_FISS dans l'opérande TABLE. Pour cela, on peut utiliser l'opérateur CALC_TABLE. Par exemple, dans le cas où la fissure est formée par deux morceaux, la combinaison des deux tableaux peut être réalisée dans la façon suivante :

```
SIF1 = CALC_G(THETA=_F(...
                                NUME_FOND= 1 ,),
                           …)
SIF2 = CALC_G(THETA=_F(...
                                NUME_FOND= 2 ,),
                           …)

SIF = CALC_TABLE(TABLE=SIF1,
                         ACTION=(_F(OPERATION='COMB',
                                    TABLE=SIF2,
                                    NOM_PARA=('NUME_ORDRE','NUME_FOND'),),
                                ),
                         );
```

Puis on utilise SIF dans l'opérateur PROPA_FISS. Ce tableau est formé par l'union des deux tableaux issus par CALC_G. Un ordre quelconque peut être utilisé dans la combinaison des tableaux.

4.7.5 Opérande NB_POINT_FOND

```
#  Si TEST_MAIL='NON'
 ◊  NB_POINT_FOND =  nbptfd,                                    [I]
```

Si l'opérande NB_POINT_FOND a été utilisé dans CALC_G pour le calcul des facteurs d'intensité des contraintes, on doit forcément utiliser cet opérande dans PROPA_FISS aussi.

L'opérateur PROPA_FISS ne peut pas vérifier si cet opérande a été utilisé ou pas dans CALC_G ou si les valeurs données sont correctes. L'utilisateur doit donc faire attention à utiliser correctement cet opérande !

L'opérande NB_POINT_FOND accepte une liste de valeurs d'entiers. Pour chaque morceau (voir opérande TABLE) de la fissure donné par FISS_ACTUELLE on doit donner une valeur de NB_POINT_FOND égale à la valeur utilisée dans CALC_G . Si l'opérande n'a pas été utilisée dans CALC_G pour un morceau, on utilise la valeur zéro. L'ordre des valeurs de NB_POINT_FOND dans la liste est donné par l'opérande NUME_FOND de CALC_G . Des exemples sont donnés à la fin de ce document.

4.8 Opérande RAYON

```
◊  RAYON  =  ray,                                              [R]
```

Cet opérande est utilisé pour définir le rayon du tore autour du fond de fissure dans lequel les résidus vont être estimés pour les étapes de réinitialisation et réorthogonalisation des level sets. Ce tore définit la zone où les level sets doivent être suffisamment proches de fonctions de distances signées.

La valeur de ce rayon doit toujours être plus grande que le rayon supérieur d'intégration de l'opérateur CALC_G. L'opérateur PROPA_FISS ne peut pas vérifier si cette condition est respectée. L'utilisateur doit donc faire attention à utiliser correctement cet opérande !

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

5 Opérandes spécifiques à la méthode MAILLAGE

5.1 Opérandes MAIL_STRUC

♦ MAIL_STRUC = mast, [maillage]

Cet opérande permet de définir le maillage de la structure saine (*i.e.* sans fissure).

5.2 Opérande ITERATION

♦ ITERATION = iter, [I]

Cet opérande permet de définir le numéro de l'itération du calcul de propagation. La première itération (propagation de la fissure initiale) est l'itération 1.

Cet opérande facilite la réalisation des calculs : il est ainsi possible de relancer un calcul de propagation sur une fissure ayant déjà été propagée *nb_init* fois.

5.3 Opérande MAIL_TOTAL

♦ MAIL_TOTAL = CO('matot2'), [TXM]

Cet opérande permet de définir le nom du maillage total, *i.e.* la concaténation du maillage de la structure saine (cf. opérande MAIL_STRUC) et du maillage de la fissure propagée MAIL_PROPAGE.

C'est ce maillage total matot2 qui doit être utilisé pour définir le modèle et la nouvelle fissure dans DEFI_FISS_XFEM.

5.4 Mot clé facteur FISSURE

Cet opérande obligatoire permet d'indiquer les fissures du modèle mo qui propagent.

5.4.1 Syntaxe FISSURE

```
♦ FISSURE      =  _F(
    ♦ MAIL_ACTUEL   =  max1                      [ maillage ]
    ♦ FISS_ACTUELLE=   FISS1                     [ fiss_xfem]
    ◊ MAIL_PROPAGE  =  CO('max2'),               [TXM]
    ◊ GROUP_MA_FOND=   /   'FOND'                [DEFAUT]
                       /   grmaff                [TXM]
    ◊ GROUP_MA_FISS=   /   'FISS'                [DEFAUT]
                       /   grmafi,               [TXM]
    ♦ TABLE         =  sif,                      [table]
```

5.4.2 Opérande MAIL_ACTUEL

♦ MAIL_ACTUEL = max1, [maillage]

Cet opérande permet de définir la fissure que l'on veut propager à l'aide de son maillage surfacique max1. Le maillage de la fissure max1 doit respecter certaines règles de nommage, cf. §6.

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

5.4.3 Opérandes FISS_ACTUELLE, TABLE

Voir les paragraphes §4.7.2 et §4.7.4.

5.4.4 Opérande MAIL_PROPAGE

```
◊  GROUP_MA_FOND=   /   'FOND'                  [DEFAUT]
                    /   grmaff                  [TXM]
◊  GROUP_MA_FISS=   /   'FISS'                  [DEFAUT]
                    /   grmafi,                 [TXM]
```

Ces opérandes permettent d'indiquer comment vont s'appeler les groupes de mailles du fond et des lèvres (qui seront successivement FOND_0 , FOND_1 , etc). Ces mots clés sont indispensables pour pouvoir utiliser un maillage quelconque pour la fissure initiale ; et pour pouvoir traiter le cas de la multi-fissuration. Une valeur par défaut existe.

5.4.5 Opérandes GROUP_MA_FOND et GROUPE_MA_FISS

```
◊  MAIL_PROPAGE =  CO('max2'),                  [TXM]
```

Cet opérande permet de définir le nom du maillage surfacique correspondant à la fissure propagée.

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

6 Opérandes spécifiques à la méthode INITIALISATION

6.1 Opérandes MAIL_STRUC

♦ MAIL_STRUC = mast, [maillage]

Cet opérande permet de définir le maillage de la structure saine (*i.e.* sans fissure).

6.2 Opérandes FORM_FISS

◆ FORM_FISS = / 'DEMI_DROITE'
/ 'DEMI_PLAN'
/ 'ELLIPSE'

Cet opérande permet de définir le type de la fissure : une demi-droite en 2D ou une fissure plane à fond rectiligne en 3D, ou une fissure elliptique en 3D.

6.3 Cas DEMI_DROITE : opérandes PFON et DTAN_DTAN

♦ PFON = (xf,yf,zf), [l_R]
♦ DTAN = (xt,yt,zt), [l_R]

Ces opérandes permettent de définir la fissure initiale en 2D en donnant la coordonnée du point du fond de fissure PFON et la direction de propagation de celle-ci DTAN.

6.4 Cas DEMI_PLAN : opérandes PFON, POINT_ORIG, POINT_EXTR et NB_POINT_FOND

♦ POINT_ORIG = (xo,yo,zo), [l_R]
♦ POINT_EXTR = (xe,ye,ze), [l_R]
♦ DTAN = (xt,yt,zt), [l_R]
♦ NB_POINT_FOND = nbfond, [I]

Ces opérandes permettent de définir la fissure initiale en 3D en donnant les coordonnées des points origine et extrémité du fond de fissure et la direction de propagation de celle-ci DTAN . Le fond de fissure sera discrétisé avec nbfond nœuds.

6.5 Cas ELLIPSE : opérandes CENTRE, DEMI_GRAND_AXE, DEMI_PETIT_AXE, VECT_X, VECT_Y, ANGLE_ORIG, ANGLE_EXTR et NB_POINT_FOND

♦ CENTRE = (xo,yo,zo), [l_R]
♦ DEMI_GRAND_AXE = a , [R]
♦ DEMI_PETIT_AXE = b, [R]
♦ VECT_X = (vxx, vxy, vxz), [l_R]
♦ VECT_Y = (vyx, vyy, vyz), [l_R]

Ces opérandes permettent de définir la fissure elliptique initiale en 3D en donnant les coordonnées du centre de l'ellipse, les valeurs des demi-petit axe et demi-grand axe, des vecteurs d'orientation du plan de l'ellipse. Pour plus de détails concernant ces mot-clés, voir U4.82.08.

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

♦ ANGLE_ORIG = ango, [R]
♦ ANGLE_EXTR = ange, [R]
♦ NB_POINT_FOND = nbpf, [I]

ANGLE_ORIG donne la position (en degrés) du premier point dans le repère (VECT_X , VECT_Y).
ANGLE_EXTR donne la position (en degrés) du dernier point dans le repère (VECT_X , VECT_Y).
NB_POINT_FOND désigne le nombre de point à créer le long du fond de fissure.

6.6 Opérandes GROUP_MA_FOND, GROUP_MA_FISS

Voir § 5.4.5

6.7 Opérandes MAIL_TOTAL

Voir § 5.3

6.8 Opérandes MAIL_FISS

Idem que MAIL_PROPAGE, voir § 5.4.4

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

7 Exemples de propagation de fissure avec X-FEM

Plusieurs cas-tests permettent de valider la propagation de fissures avec X-FEM.

En 2D :

- sslp318 : propagation d'une fissure X-FEM non débouchante sollicitée en mode ;
- sslp319 : propagation de deux fissures X-FEM débouchantes sollicitée en mode ;
- sslp318 : Propagation d'une fissure X-FEM non débouchante sollicitée en mode ;
- sslp320 : Propagation d'une fissure X-FEM débouchante sollicitée en Mode I ;
- sslp321 : Propagation d'une fissure X-FEM dans une plaque en flexion 3 points.

En 3D :

- sslv315 : propagation en 3D d'une fissure inclinée ;
- sslv314 : Propagation plane d'une fissure débouchante avec X-FEM ;
- sslv316 : Fissuration à propagation imposée avec X-FEM ;
- sslv320 : Propagation plane d'une fissure 3D se divisant et fusionnant avec X-FEM.

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