
Opérateur DEFI_FISS_XFEM

1 But

Définir une fissure ou un interface 2D ou 3D par le biais de fonctions de niveaux.

Pour les fissures, on définit deux fonctions de niveaux (level sets) permettant de caractériser une fissure quelconque (fissure plane ou non). Pour les interfaces, une seule level-set est nécessaire. Les level sets peuvent être définies de trois façons différentes :

- 1) soit à partir de deux groupes de mailles (le groupe de mailles de la fissure et le groupe de mailles du fond de fissure) ;
- 2) soit à partir de deux fonctions analytiques.
- 3) soit à partir d'un catalogue de formes prédéfinies.

Pour le cas 2D, les mailles du fond de fissure sont des mailles `POI` ; elles ne doivent pas être données en tant que groupe de nœud mais en tant que groupe de mailles. Pour le cas 3D, l'orientation du fond de fissure peut être réalisée soit automatiquement soit manuellement. Ce deuxième cas de figure nécessite la donnée d'un point initial et d'un plan (vecteur normal et un point du plan).

Les caractéristiques liées au contact sur les lèvres de la fissure ne sont pas données ici mais dans une charge de contact spécifique avec l'opérateur `DEFI_CONTACT`.

L'aspect théorique de la méthode X-FEM est abordée dans [R7.02.12], et il est conseillé à l'utilisateur qui n'est pas familier avec l'usage des level sets de lire les quelques lignes explicatives relatives à la représentation de surfaces par la méthode des level sets.

De plus, cet opérateur s'utilise aussi avec des fissures maillées, en préalable au calcul des facteurs d'intensité de contraintes en 3d par l'opérateur `CALC_G` [U4.82.03] avec l'option `CALC_K_G`.

L'opérateur produit un concept de type `fiss_xfem`.

2 Syntaxe

```
[fiss_xfem] = DEFI_FISS_XFEM(

    ♦ MODELE = mo, [modele]

# choix d'une fissure ou d'une interface
    ◊ TYPE_DISCONTINUITE = / 'FISSURE', [DEFAULT]
                        / 'INTERFACE'

# Définition de la fissure ou de l'interface

    ♦ DEFI_FISS = _F (

# Soit on définit le groupe de mailles d'une lèvres et le groupe de mailles
du fond de fissure :

    ♦ / ♦ GROUP_MA_FISS = grma, [l_gr_maille]
        ◊ GROUP_MA_FOND = grma, [l_gr_maille]

# Soit on définit deux fonctions analytiques ; une pour caractériser la
surface de la fissure, et l'autre pour caractériser le fond de fissure :

    / ♦ FONC_LN = fonc, [fonction]
        ◊ FONC_LT = fonc, [fonction]

# Soit on utilise une forme de fissure ou d'interface pré-définie et on
donne les caractéristiques géométriques :

    / ♦ FORM_FISS = form, [K8]

    # Si form = 'ELLIPSE'
    ♦ DEMI_GRAND_AXE = a, [R]
    ♦ DEMI_PETIT_AXE = b, [R]
    ♦ CENTRE = (x0, y0, z0), [l_R]
    ♦ VECT_X = (vxx, vxy, vxz), [l_R]
    ♦ VECT_Y = (vyx, vyy, vyz), [l_R]
    ◊ COTE_FISS = / 'IN', [DEFAULT]
                / 'OUT',

    # Si form = 'RECTANGLE'
    ♦ DEMI_GRAND_AXE = a, [R]
    ♦ DEMI_PETIT_AXE = b, [R]
    ♦ RAYON_CONGE = r, [R]
    ♦ CENTRE = (x0, y0, z0), [l_R]
    ♦ VECT_X = (vxx, vxy, vxz), [l_R]
    ♦ VECT_Y = (vyx, vyy, vyz), [l_R]
    ◊ COTE_FISS = / 'IN', [DEFAULT]
                / 'OUT',

    # Si form = 'CYLINDRE' ou 'INCLUSION'
    ♦ DEMI_GRAND_AXE = a, [R]
    ♦ DEMI_PETIT_AXE = b, [R]
    ♦ CENTRE = (x0, y0, z0), [l_R]
    ♦ VECT_X = (vxx, vxy, vxz), [l_R]
    ♦ VECT_Y = (vyx, vyy, vyz), [l_R]

    # Si form = 'DEMI_PLAN'
    ♦ PFON = (x0, y0, z0), [l_R]
    ♦ NORMALE = (vnx, vny, vnz), [l_R]
    ♦ DTAN = (vtx, vty, vtz), [l_R]
```

```
# Si form = SEGMENT'
♦ PFON_ORIG      = (x0, y0, z0),      [1_R]
♦ PFON_EXTR      = (x1, y1, z1),      [1_R]

# Si form = 'DEMI_DROITE'
♦ PFON           = (x0, y0, z0),      [1_R]
♦ DTAN           = (vtx, vty, vtz),    [1_R]

# Si form = 'DROITE'
♦ POINT          = (x0, y0, z0),      [1_R]
♦ DTAN           = (vtx, vty, vtz),    [1_R]

),

# Définition de la zone d'enrichissement

◇ GROUP_MA_ENRI   = grma,              [1_gr_maille]
◇ TYPE_ENRI_FOND  = / 'TOPOLOGIQUE'     [DEFAULT]
                  / 'GEOMETRIQUE'

# Si TYPE_ENRI_FOND = 'GEOMETRIQUE'
◇ RAYON_ENRI      = Renri,              [R]

# Si RAYON_ENRI = None
◇ NB_COUCHES      = / 4,                [DEFAULT]
                  = / N,                [I]

# Orientation du fond de fissure

◇ ORIE_FOND       = _F (

    ♦ PFON_INI     = (Pix , Piy , Piz),  [1_R]
    ♦ VECT_ORIE    = (Vox , Voy , Voz),  [1_R]
    ♦ POINT_ORIG   = (Pox , Poy , Poz),  [1_R]

),

# Impression d'informations

◇ INFO            = / 1,                [DEFAULT]
                  / 2,
                  / 3,

)
```

3 Opérandes

3.1 Opérande **MODELE**

♦ `MODELE = mo`

`mo` : nom du modèle sur lequel on va définir la fissure.

La définition d'une fissure par cet opérateur n'est autorisée que pour les modélisations suivantes :

- 1) `3D, C_PLAN, D_PLAN`, pour n'importe quel type de mailles.
- 2) `3D_XFEM_CONT, C_PLAN_XFEM_CONT, D_PLAN_XFEM_CONT`, ces modélisation servant exclusivement pour traiter le contact entre les lèvres de la fissures avec la formulation du contact dite « aux arêtes »

3.2 Opérande **TYPE_DISCONTINUITE**

♦ `TYPE_DISCONTINUITE = / 'FISSURE', [DEFAULT]
/ 'INTERFACE'`

Par défaut, on définit une fissure. Il est cependant possible de définir une interface (pour traiter le cas de 2 solides en contact, ou des sous-épaisseurs, ...)

3.3 Mot clé **DEFI_FISS**

♦ `DEFI_FISS = _F`

Le mot-clé facteur `DEFI_FISS` permet de définir une fissure ou une interface de trois manières différentes :

- soit à partir de deux groupes de mailles (voir §4) ;
- soit à partir de deux fonctions analytiques (voir §5) ;
- soit à partir d'un catalogue de formes prédéfinies (voir §5).

3.3.1 Définition de la fissure par des groupes de mailles

Si on a une fissure (ou une interface) déjà maillée, alors on peut définir la fissure en donnant deux groupe de mailles : `GROUP_MA_FISS` et `GROUP_MA_FOND`. Ce maillage peut être indépendant du maillage de la structure.

/ ♦ `GROUP_MA_FISS = grma`
avec `grma` groupe de mailles unique.

Ce groupe de mailles correspond à une seule des lèvres de la fissure (lèvre inférieure ou supérieure). Ce groupe de mailles doit être orienté. Dans le cas de lèvres qui ne seraient pas parfaitement collées (entaille), le fait de privilégier un côté aura une légère influence sur la base locale du fond de fissure, influence d'autant plus grande que l'angle entre les deux lèvres est important. Pour le cas 2D, le programme trie les mailles de `GROUP_MA_FISS` de façon à avoir un groupe de mailles contiguës quel que soit le groupe de départ. Il n'est donc pas nécessaire de donner un groupe de mailles qui se suivent.

♦ `GROUP_MA_FOND = grma`
avec `grma` groupe de mailles unique.

Cet opérande sert à définir le fond de fissure. Pour les interfaces, `GROUP_MA_FOND` n'est pas à renseigner.

En 3D, ce groupe de mailles peut être un groupe de mailles fermé ou ouvert. Il est obligatoire que ce groupe contienne des mailles continues, c'est-à-dire que deux mailles qui se suivent aient un

nœud en commun. Sinon, il est possible que l'orientation du fond de fissure soit fausse. Un message d'erreur fatale prévient l'utilisateur dans le cas où la phase d'orientation a échoué.

En 2d, GROUP_MA_FOND est un groupe de mailles point.

3.3.2 Définition de la fissure par des fonctions analytiques

Le principal intérêt de cet opérateur est la possibilité de définir une fissure sans que celle-ci ne soit forcément maillée. Dans ce cas, on définit la fissure à l'aide de deux fonctions de niveaux (level sets). La première level set (dite level set « normale ») est celle qui permet de caractériser la surface de la fissure. On renseigne donc FONC_LN avec une fonction réelle de X , Y et Z définie au préalable par l'opérateur FORMULE. La surface de la fissure sera alors l'iso-zéro de cette fonction. La seconde level set (dite level set « tangente ») est celle qui permet de caractériser le fond de fissure. On renseigne donc FONC_LT avec une fonction réelle de X , Y et Z définie au préalable par l'opérateur FORMULE. La trace de l'iso-zéro de FONC_LT dans le plan de fissuration est le fond de fissure. Les points de la fissure sont alors caractérisés par $\text{FONC_LN} = 0$ et $\text{FONC_LT} < 0$, alors que le fond de fissure est caractérisé par $\text{FONC_LN} = 0$ et $\text{FONC_LT} = 0$.

Pour les interfaces, FONC_LT n'est pas à renseigner.

```
/ ♦ FONC_LN          =  fonc,
    avec fonc une fonction ou une formule définie auparavant.

♦ FONC_LT          =  fonc,
    avec fonc une fonction ou une formule définie auparavant.
```

Remarque :

Pour une fissure, il est primordial que les level sets sont les vraies fonctions distances, on ne peut donc pas définir une fissure elliptique par :
 $\text{LN}=\text{FORMULE}(\text{NOM_PARA}=('X','Y','Z'),\text{VALE}='Z-H')$
 $\text{LT}=\text{FORMULE}(\text{NOM_PARA}=('X','Y','Z'),\text{VALE}='(X/a)^2+(Y/b)^2-1')$
où a et b seraient les demi-grand axe et demi-petit axe de l'ellipse. En effet, la fonction LT n'est pas la distance à l'ellipse. La fonction distance à une ellipse ne s'exprime pas à l'aide de fonctions usuelles. Pour introduire une telle fissure, il faut utiliser la 3^{ème} méthode : les formes pré-définies.
Pour les interfaces, n'importe quelle fonction level set est possible, et il n'est pas utile d'introduire la vraie fonction distance.

Remarque :

Attention au signe de FONC_LT, toujours se rappeler que la fissure est définie du côté $\text{FONC_LT} < 0$.

3.3.3 Définition de la fissure par un catalogue de forme pré-définies

Il existe un catalogue de formes de fissure/interface pré-définies :

- pour les fissures : ELLIPSE, RECTANGLE, CYLINDRE, DEMI_PLAN, SEGMENT, DEMI_DROITE,
- pour les interfaces : INCLUSION, DROITE.

Pour chacune, il faut donner les caractéristiques géométriques spécifiques.

Un des avantages de prendre une fissure du catalogue est que en 3d l'orientation du fond de fissure est faite de manière automatique (voir §9).

```
/ ♦ FORM_FISS      =  form,

si form = ELLIPSE
    Cas d'une fissure plane elliptique tridimensionnelle. Il faut renseigner la longueur du demi-
    grand axe par le mot-clé DEMI_GRAND_AXE et la longueur du demi-petit axe par le mot-clé
    DEMI_PETIT_AXE. Pour caractériser le plan de l'ellipse, on doit renseigner le centre de
```

l'ellipse par le mot-clé `CENTRE` et le repère du plan. Le demi-grand axe correspond au vecteur `VECT_X` et le demi-petit axe au vecteur `VECT_Y`. De plus, on peut spécifier de quel coté se trouve la fissure (à l'intérieur ou à l'extérieur de l'ellipse) par le mot-clé `COTE_FISS`.

- ♦ `DEMI_GRAND_AXE` = `a`,
- ♦ `DEMI_PETIT_AXE` = `b`,
- ♦ `CENTRE` = `(x0, y0, z0)`,
- ♦ `VECT_X` = `(vxx, vxy, vxz)`,
- ♦ `VECT_Y` = `(vyx, vyy, vyz)`,
- ♦ `COTE_FISS` = `/ 'IN',`
`/ 'OUT',`

si `form = RECTANGLE`

Cas d'une fissure de forme rectangulaire, avec possibilité de coins arrondis. C'est une forme similaire à l'ellipse, il faut renseigner les mêmes mot-clés : la valeur du demi-grand axe (ou demi-longueur) par le mot-clé `DEMI_GRAND_AXE` et la valeur du demi-petit axe (ou demi-largeur) par le mot-clé `DEMI_PETIT_AXE`. Pour caractériser le plan du rectangle, on doit renseigner le centre du rectangle par le mot-clé `CENTRE` et le repère du plan. Le demi-grand axe correspond au vecteur `VECT_X` et le demi-petit axe au vecteur `VECT_Y`. De plus, on peut spécifier de quel coté se trouve la fissure (à l'intérieur ou à l'extérieur du rectangle) par le mot-clé `COTE_FISS`. Le seul mot-clé en plus est celui servant à donner la valeur du rayon du congé pour les coins arrondis (`RAYON_CONGE`) :

- ♦ `DEMI_GRAND_AXE` = `a`,
- ♦ `DEMI_PETIT_AXE` = `b`,
- ♦ `RAYON_CONGE` = `r`,
- ♦ `CENTRE` = `(x0, y0, z0)`, #coordonnées du point C
- ♦ `VECT_X` = `(vxx, vxy, vxz)`, #vecteur e1
- ♦ `VECT_Y` = `(vyx, vyy, vyz)`, #vecteur e2
- ♦ `COTE_FISS` = `/ 'IN',`
`/ 'OUT',`

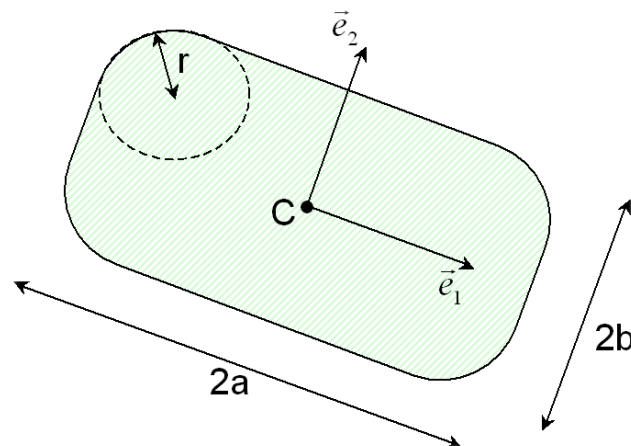


Figure 3.3.3-a: schéma d'une fissure rectangulaire à coins arrondis

si `form = CYLINDRE`

Cas d'une fissure tridimensionnelle cylindrique à section elliptique. Pour caractériser le fond (elliptique) de la fissure, il faut renseigner la longueur du demi-grand axe par le mot-clé `DEMI_GRAND_AXE` et la longueur du demi-petit axe par le mot-clé `DEMI_PETIT_AXE`. On doit également renseigner le centre de l'ellipse par le mot-clé `CENTRE` ainsi que le repère du plan (`VECT_X`, `VECT_Y`). Remarque : le demi-grand axe correspond au vecteur `VECT_X` et le demi-petit axe au vecteur `VECT_Y`. La génératrice du cylindre et la direction de propagation potentielle de la fissure sont données par le produit vectoriel de `VECT_X` par `VECT_Y`. Ce cylindre est semi-infini suivant cet axe.

```
♦ DEMI_GRAND_AXE = a,  
♦ DEMI_PETIT_AXE = b,  
♦ CENTRE         = (x0, y0, z0),  
♦ VECT_X         = (vxx, vxy, vxz),  
♦ VECT_Y         = (vyx, vyy, vyz),
```

si form = DEMI_PLAN

Cas d'une fissure définie par un demi-plan. PFON désigne un point du fond de fissure. NORMALE définit la normale au demi-plan de la fissure et DTAN fournit un vecteur dans le plan de fissure, orthogonale au fond et dirigé dans la direction de propagation potentielle.

```
♦ PFON           = (x0, y0, z0),  
♦ NORMALE        = (vnx, vny, vnz),  
♦ DTAN           = (vtx, vty, vtz),
```

si form = SEGMENT

Cas d'une fissure 2D définie par un segment. PFON_ORIG et PFON_EXTR désignent les extrémités du segment.

```
♦ PFON_ORIG      = (x0, y0, z0),  
♦ PFON_EXTR      = (x1, y1, z1),
```

si form = DEMI_DROITE

Cas d'une fissure 2D définie par une demi-droite. PFON désigne le point du fond de fissure. DTAN correspond à un vecteur directeur de la demi-droite orienté dans la direction de propagation potentielle.

```
♦ PFON           = (x0, y0, z0),  
♦ DTAN           = (vtx, vty, vtz),
```

si form = INCLUSION

Cas d'une inclusion (interface) dans une structure 2D. Cela sert à représenter un trou (du vide) ou une sous-épaisseur, ou un affouillement, ou bien encore une interface entre bimatériau. Les mot-clés sont les mêmes que pour une fissure cylindrique :

```
♦ DEMI_GRAND_AXE = a,  
♦ DEMI_PETIT_AXE = b,  
♦ CENTRE         = (x0, y0, z0),  
♦ VECT_X         = (vxx, vxy, vxz),  
♦ VECT_Y         = (vyx, vyy, vyz),
```

si form = DROITE

Cas d'une interface droite dans une structure 2D définie. POINT désigne un point de l'interface. DTAN correspond à un vecteur directeur de la droite. L'orientation n'est pas importante.

```
♦ POINT          = (x0, y0, z0),  
♦ DTAN           = (vtx, vty, vtz),
```

Exemples :

- 1) si on souhaite définir un barreau fissuré de part en part dans son plan à la hauteur $z = H$ (voir [Figure 3.2-a], sur laquelle la fissure est hachurée) :

```
LN=FORMULE(NOM_PARA=('X','Y','Z'),VALE='Z-H');  
LT=FORMULE(NOM_PARA=('X','Y','Z'),VALE='Y-a');
```

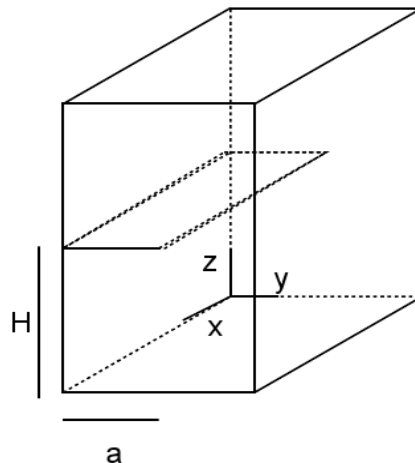


Figure 3.2-a : Barreau fissuré

- 2) si on souhaite définir une fissure elliptique dans un massif infini dans un plan à la hauteur $z = H$ (voir [Figure 3.2-b], sur laquelle la fissure est hachurée) :

```
DEFI_FISS = _F ( FORM_FISS      = 'ELLIPSE',
                  DEMI_GRAND_AXE  = a,
                  DEMI_PETIT_AXE  = b,
                  CENTRE          = (0, 0, H),
                  VECT_X          = (1, 0, 0),
                  VECT_Y          = (0, 1, 0),
                  COTE_FISS       = 'IN', )
```

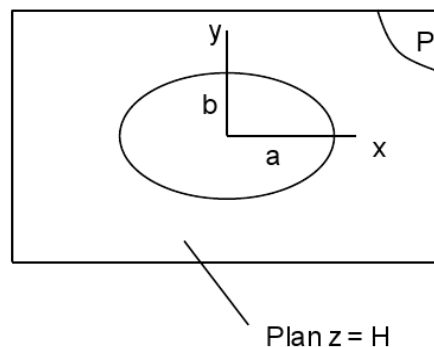


Figure 3.2-b : Fissure elliptique

3.4 Opérande GROUP_MA_ENRI

◇ GROUP_MA_ENRI = grma, [l_gr_maille]

Cette opérande permet de connaître la zone sur laquelle la procédure d'enrichissement va s'effectuer. Les nœuds enrichis doivent appartenir à ce groupe de maille.

Si GROUP_MA_ENRI n'est pas renseigné, alors la procédure d'enrichissement va s'effectuer sur tout le maillage associé au modèle.

Une utilisation de ce mot-clé permet donc de délimiter la zone d'enrichissement, donc de la fissure. L'exemple suivant (voir [Figure 3.3-3.4-a]) présente une fissure de forme elliptique dans un cylindre creux. La fissure est définie par la donnée de groupes de mailles (mots-clé GROUP_MA_FISS et GROUP_MA_FOND). Deux level sets en sont déduites, en considérant un prolongement du fond de fissure par continuité. Si la zone d'enrichissement contient toute la structure, cela conduit alors à une fissure dans la partie droite du cylindre, ce qui n'est pas du tout ce que l'on souhaite. Pour résoudre ce

problème, il suffit de limiter la zone d'enrichissement et de renseigner le mot-clé `GROUP_MA_ENRI` avec les mailles de la moitié gauche du cylindre par exemple.

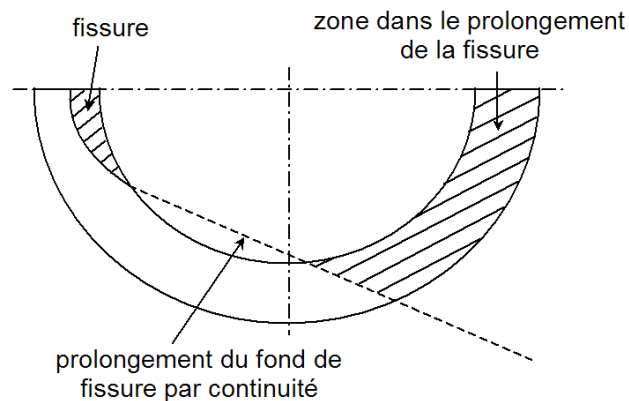


Figure 3.3-3.4-a : Coupe horizontale dans le plan de la fissure

3.5 Opérandes `TYPE_ENRI_FOND`, `RAYON_ENRI`, `NB_COUCHES`

```
◇ TYPE_ENRI_FOND = / 'TOPOLOGIQUE' [DEFAULT]
                  / 'GEOMETRIQUE'
```

Ce mot-clé permet de choisir le type d'enrichissement en fond de fissure. On active l'enrichissement topologique (une seule couche) si `TYPE_ENRI_FOND` = 'TOPOLOGIQUE' (par défaut). On active l'enrichissement géométrique (plusieurs couches) si `TYPE_ENRI_FOND` = 'GEOMETRIQUE' (pour plus de renseignements concernant les effets de l'enrichissement sur la qualité des résultats, voir [R7.02.12] et [U2.05.02]). L'enrichissement géométrique est piloté par les mots-clé `RAYON_ENRI` et `NB_COUCHES`. Ce type d'enrichissement n'est pas du tout conseillé avec des maillages quadratiques (sous peine de pivot nul).

```
◇ RAYON_ENRI = Renri, [R]
```

Ce mot-clé ne peut être renseigné que si `TYPE_ENRI_FOND` = 'GEOMETRIQUE'. Si cette opérande est renseignée, il permet de préciser un critère géométrique `Renri` tel que tous les nœuds dont la distance au fond de fissure est inférieure à ce critère seront enrichis par les fonctions singulières. Cet enrichissement est appelé « géométrique ». Des études ont montré qu'un tel enrichissement améliorerait grandement la précision des calculs, notamment lorsque le maillage est raffiné en fond de fissure. Il est conseillé de choisir `Renri` environ égal à 1/10 de la longueur de la fissure. L'inconvénient est qu'un trop grand rayon d'enrichissement peut détériorer le conditionnement du système et engendrer des difficultés de convergence.

Pour éviter les désagréments causés par un trop grand rayon d'enrichissement, on propose de définir la zone à enrichir en stipulant le nombre de couches d'éléments à enrichir autour du fond de fissure à la place du rayon d'enrichissement.

```
# Si RAYON_ENRI = None
◇ NB_COUCHES = / 4, [DEFAULT]
               / N, [I]
```

Si `RAYON_ENRI` n'est pas renseigné (et que `TYPE_ENRI_FOND` = 'GEOMETRIQUE'), alors `NB_COUCHES` est le nombre de couches d'éléments à enrichir (4 par défaut). Généralement, 7 couches est la limite à ne pas dépasser pour éviter les pivots nuls lors de la résolution.

3.6 Mot clé `ORIE_FOND`

Deux façons d'orienter le fond de fissure sont possibles en 3D :

- méthode utilisateur (avec `ORIE_FOND`),
- méthode automatique (sans `ORIE_FOND`, uniquement si la fissure est définie à partir du catalogue de formes : `FORM_FISS`).

Il ne faut pas renseigner `ORIE_FOND` pour les interfaces.

3.6.1 Méthode utilisateur pour l'orientation du fond

```

◇  ORIE_FOND      =  _F (
    ♦  PFON_INI      =  (Pix , Piy , Piz),          [1_R]
    ♦  VECT_ORIE     =  (Vox , Voy , Voz),          [1_R]
    ♦  POINT_ORIG    =  (Pox , Poy , Poz),          [1_R]

```

Le mot clé `ORIE_FOND`, nécessaire uniquement en 3D, permet d'orienter le fond de fissure. En effet, grâce aux level sets, on peut déterminer des points du fond de fissure (qui sont les intersections des faces des mailles volumiques avec les iso-zéros des 2 level-sets). On a donc une liste de points de l'espace à ordonner. Pour cela, on détermine le point initial du fond de fissure comme étant le point le plus proche de `PFON_INI`. Ensuite, on projette orthogonalement les points du fond sur le plan de normale `VECT_ORIE` passant par `POINT_ORIG`. L'orientation des points se fait ensuite naturellement, en partant du point initial et en tournant dans le sens trigonométrique autour de `VECT_ORIE` (voir exemples ci-dessous).

Comment bien choisir ces trois paramètres ?

- 1) `PFON_INI` : Point initial du fond de fissure défini le plus précisément possible par ses coordonnées. Dans le cas d'un fond fermé, il faut quand même choisir un point initial (on se ramène dans tous les cas à un fond ouvert). Si le `PFON_INI` fournit n'est pas un nœud du bord (et qu'il existe des nœuds de bord), on émet un message d'alarme et on réalise une orientation automatique (voir partie suivante).
- 2) `VECT_ORIE` : Dans le cas d'une fissure plane ou presque plane, il est préférable de prendre la normale (ou la normale approchée) au plan de fissure.
- 3) `POINT_ORIG` : Ce point ne doit pas se trouver sur le fond de fissure ni sur la projection du fond suivant `VECT_ORIE`. Souvent, le centre de la fissure est un bon choix.

Exemples : Reprenons les deux exemples de fissures cités ci-dessus.

- 1) barre fissuré :

Ici, si l'on veut définir le fond comme le segment $[AB]$ (voir [Figure 3.4-a]), alors on peut choisir :

<code>PFON_INI</code>	point A
<code>VECT_ORIE</code>	axe z
<code>POINT_ORIG</code>	point C

En effet, le choix de `VECT_ORIE` et de `POINT_ORIG` implique que l'on se place dans le plan (A, B, C) . Imaginons que l'on ait plusieurs points à orienter [Figure 3.4-b], on vérifie alors que dans un tel cas, le fond de fissure sera bien orienté de A vers B :

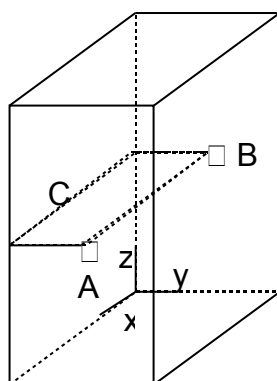


Figure 3.4-a : Barreau fissuré

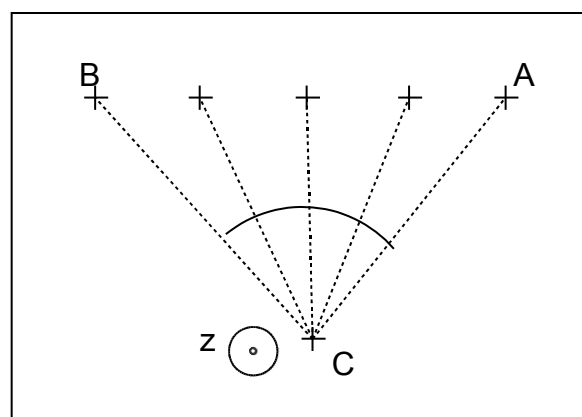


Figure 3.4-b : Plan de fissure

On remarque que si on avait choisi `VECT_ORIE` dans le sens opposé, on aurait eu la flèche dans le sens opposé, et donc les points du fond auraient été : A , puis B , puis le point juste à droite de B , et ainsi de suite...

2) fissure elliptique :

Ici, si l'on désire définir le fond comme le chemin partant de A et parcourant l'ellipse dans le sens trigonométrique [voir Figure 3.4-c], alors on peut choisir :

<code>PFON_INI</code>	point A
<code>VECT_ORIE</code>	axe z
<code>POINT_ORIG</code>	point $O : (0,0,H)$

En effet, le choix de `VECT_ORIE` et de `POINT_ORIG` implique que l'on se place dans le plan $z = H$. Imaginons que l'on ait plusieurs points à orienter [Figure 3.4-c], on vérifie alors que le fond de fissure sera bien orienté en partant de A et en parcourant l'ellipse dans le sens trigonométrique.

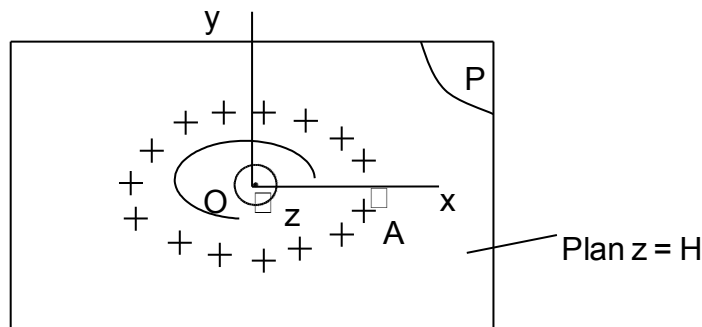


Figure 3.4-c : Orientation d'une fissure elliptique

Conseil :

Toujours bien vérifier que la liste ordonnée des points du fond de fissure correspond bien à celle que l'on attend. Cette liste est automatiquement affichée lors du passage dans la commande `DEFI_FISS_XFEM`.

3.6.2 Méthode automatique pour l'orientation du fond

La définition des paramètres pour l'orientation du fond de fissure : `ORIE_FOND` peut s'avérer problématique pour des maillages complexes. Ce mot clé est optionnel (orientation automatique) en l'absence de celui-ci. La méthode utilisée est identique à celle présentée précédemment. Elle s'appuie sur les paramètres entrés par l'utilisateur pour définir la fissure avec `FORM_FISS`.

Remarque :

L'orientation automatique n'est possible que si l'utilisateur a défini la fissure à partir du catalogue de formes prédéfinies `FORM_FISS`.

Pour information on donne les valeurs automatiques de `POINT_ORIG` et `VECT_ORIE` (`PFON_INI` devient inutile) à partir des informations géométriques entrées par l'utilisateur avec `FORM_FISS` :

ELLIPSE, RECTANGLE ou CYLINDRE :

`POINT_ORIG` = CENTRE (centre de l'ellipse)

`VECT_ORIE` = `VECT_X` . `VECT_Y` (produit vectoriel des axes de l'ellipse)

DEMI_PLAN :

`POINT_ORIG` = `PFON` - `DTAN` (point du fond - vecteur direction de propagation)

`VECT_ORIE` = NORMALE (normale au plan de fissure)

Principe de l'orientation automatique :

On part de n'importe quel point du fond et on ordonne les nœuds avec la méthode présentée précédemment. On effectue ensuite une permutation circulaire des nœuds ordonnés en commençant par le dernier nœud de bord.

3.7 Opérande INFO

- / 1 : impression sur le fichier 'MESSAGE'
 - 1) du nombre de points du fond de fissure ;
 - 2) du nombre de fonds de fissure ;
 - 3) de la liste ordonnée des coordonnées des points du fond ainsi que leur abscisse curviligne, pour chaque fond de fissure.
- / 2 : mêmes impressions que 1 + impressions
 - 1) du nombre de level sets réajustées ;
 - 2) du nombre de mailles de la zone de la fissure ;
 - 3) du nombre de mailles contenant le fond de fissure
 - 4) du nombre de mailles de type Heaviside, Crack-tip et Heaviside Crack-tip.

- / 3 : même impression que 2 + impressions
- 5) des étapes du calcul ;
 - 6) des champs level sets et de leurs gradients ;
 - 7) du statut des nœuds ;
 - 8) des numéros des mailles enrichies.

4 Exemples

4.1 Avec des formules analytiques

```
LN = FORMULE ( NOM_PARA=('X','Y','Z'),VALE= 'Z-12.5')
LT = FORMULE ( NOM_PARA=('X','Y','Z'),VALE= 'X-10'.)

FISS= DEFI_FISS_XFEM ( MODELE = MODELEIN,
                      DEFI_FISS =_F( FONC_LT = LT,
                                      FONC_LN = LN, ),
                      GROUP_MA_ENRI= 'VOL',
                      ORIE_FOND=_F( PFON_INI = (2.5, 0., 12.5),
                                      VECT_ORIE = (0., 0., 1.),
                                      POINT_ORIG = (-0.5, 0., 1.5) ),),
                      INFO=2,
                      )
```

4.2 Avec des groupes de mailles

```
FISS= DEFI_FISS_XFEM ( MODELE = MODELEIN,
                      DEFI_FISS =_F( GROUP_MA_FISS = 'LEVSUP',
                                      GROUP_MA_FOND = 'FONFIS',),
                      GROUP_MA_ENRI = 'VOL',
                      ORIE_FOND=_F( PFON_INI = (0., -2., 0.),
                                      VECT_ORIE = (0., 0., -1.),
                                      POINT_ORIG = (2., 0., 0.) ),),
                      INFO=3,
                      )
```

4.3 Avec le catalogue de formes prédéfinies

Avec orientation du fond utilisateur :

```
FISS=DEFI_FISS_XFEM(MODELE=MODELEIN,
                    DEFI_FISS=_F(
                                FORM_FISS = 'CYLINDRE',
                                DEMI_GRAND_AXE = r,
                                DEMI_PETIT_AXE = r,
                                CENTRE = (0., l_fiss, 0.),
                                VECT_X = (0., 0., 1.),
                                VECT_Y = (1., 0., 0.),
                                ),
                    GROUP_MA_ENRI='VOL',
                    ORIE_FOND=_F(PFON_INI = (XFONI, YFONI, ZFONI),
                                VECT_ORIE = (XVECT, YVECT, ZVECT),
                                POINT_ORIG = (XORI, YORI, ZORI)),
                    INFO=1,
                    );
```

Avec orientation du fond automatique :

```
FISS=DEFI_FISS_XFEM(MODELE=MODELEIN,
                    INFO=1,
                    DEFI_FISS=_F(
                                FORM_FISS = 'DEMI_PLAN',
                                PFON = (0., 5., 15.),
                                NORMALE = (0., 0., 1.),
                                DTAN = (0., -1., 0.)),
                    GROUP_MA_ENRI='VOL');
```