Notice d'utilisation des modèles de zones cohésives

Résumé

Cette documentation est destinée à fournir les informations nécessaires à un utilisateur souhaitant modéliser l'amorçage et la propagation de fissure avec les modèles cohésifs (CZM).

Dans une première partie on rappelle les grandes lignes de ce type de modèles, leurs domaines d'application, avantages et limitations. La deuxième partie est destinée à fournir des recommandations spécifiques pour la réalisation du maillage. Des conseils d'utilisation dans différents domaines de la mécanique de la rupture sont détaillés dans la troisième partie. Enfin, on termine cette documentation par un exemple de fichier de commande commenté en faisant référence aux conseils prodigués dans les parties précédentes.

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

Table des Matières

1 Introduction....4
2 Synthèse des modèles cohésifs....5
2.1 Cadre théorique....5
2.2 Modèles éléments finis....5
2.3 Lois cohésives....6
2.4 Définition du repère local à la fissure....7
3 Conseils sur le maillage....9
3.1 Généralités....9
3.2 Création du maillage....9
3.2.1 Outils Code_Aster....10
3.2.2 Conseils GMSH....11
3.2.3 Conseils Salomé....12
3.2.4 Cas des configurations géométriques avancées....12
3.3 Modélisation de la zone cohésive....12
3.4 Définition d'une pré-fissure cohésive....13
3.5 Multi-fissuration....13
4 Conseils d'utilisation....15
4.1 Généralités....15
4.1.1 Paramètres mécaniques....15
4.1.2 Paramètres numériques....15
4.1.3 Chargements et conditions aux limites....16
4.1.4 Critère de convergence....16
4.1.5 Solveur linéaire....17
4.2 Une interface linéaire pour valider sa mise en données....17
4.3 Calculs en rupture fragile....17
4.3.1 Matériau linéaire....17
4.3.2 Matériau non linéaire....17
4.3.3 Outils numériques....18
4.4 Calculs en rupture ductile....20
4.4.1 Outils numériques....20
4.5 Calculs en fatigue....21
4.6 Calculs en dynamique....22
4.7 Post-traitement de la fissuration et commande POST_CZM_FISS....22
5 Mise en œuvre d'un calcul CZM....24
5.1 Liste des cas tests avec les modèles cohésifs....24
5.2 Exemple commenté, poutre DCB en fatigue....25
6 Conclusion....30
7 Bibliographie....31

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

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

1 Introduction

L'ambition des modèles cohésifs est de représenter l'évolution des surfaces d'endommagement et/ou de rupture d'un solide en 2D ou 3D, initialement sain ou déjà partiellement fissuré, soumis à une gamme de sollicitations très large (mode mixte, chargements cycliques, thermiques…) en régime quasi-statique ou dynamique pour un matériau présentant ou non des non linéarités dans son comportement (élasticité, plasticité, viscosité, ...). Cette ambition est toutefois limitée, pour l'instant, à une évolution des surfaces dont la direction est connue *a priori* .

Cette approche, largement développée dans la littérature ces dernières années, est attractive pour décrire les phénomènes de fissuration. Elle est intermédiaire entre les critères d'amorçage (taux de restitution d'énergie ou critères en contrainte) et la mécanique continue de l'endommagement. Lorsque la localisation géométrique des fissures potentielles est connue, elle fournit un cadre sûr et robuste pour prédire la cinétique de propagation.

L'objectif de cette documentation est de fournir un guide d'utilisation aux utilisateurs. D'une part aux unités de recherche dans le but de maintenir ces modèles et les faire évoluer ; mais également pour faciliter leur utilisation par l'ingénierie. Dans une première partie on rappelle succinctement les grandes lignes des modèles disponibles dans *Code_Aster*, leurs avantages et limites respectifs ainsi que les références nécessaires à une compréhension plus fine de leur formulation. Les précautions à prendre lors de la réalisation du maillage sont exposées dans la deuxième partie. La troisième est dédiée aux conseils d'utilisation dans les différents domaines de la mécanique de la rupture. Enfin, on termine cette documentation par un exemple d'utilisation commenté.

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

2 Synthèse des modèles cohésifs

2.1 Cadre théorique

Le modèles cohésifs présentés ici sont basés sur l'approche variationnelle de la rupture proposée par Francfort et Marigo [bib2]. Fondée sur un principe de minimisation d'énergie, elle permet de représenter l'amorçage et la propagation de fissures, sans hypothèses a priori sur leurs évolutions spatio-temporelles et quel que soit le chargement appliqué. Cette théorie postule que la configuration d'équilibre d'une structure est celle qui minimise, à chaque incrément de chargement, son énergie totale, somme de l'énergie potentielle et de l'énergie de surface. Cette dernière correspond à l'énergie dissipée pour créer une ou plusieurs fissures.
Les modèles numériques proposés ici sont limités à la recherche d'état d'équilibre (ou satisfaisant aux équations du mouvement en dynamique) d'une structure pour laquelle on connait au préalable le trajet de fissuration potentiel.

2.2 Modèles éléments finis

Il existe dans Code_Aster trois type d'éléments finis destinés à modéliser la fissuration avec une loi cohésive. Les deux premiers présentés ci-dessous sont classiquement utilisés dans littérature (voir [bib5] ou [bib6]), le troisième est novateur et permet de conjuguer les avantages des deux précédents même si, pour le moment, il reste en cours de validation.

L'élément de joint (EJ) classique, disposé le long des fissures potentielles, permet de discrétiser le saut de déplacement avec un ordre conforme aux autres éléments du maillage. En contrepartie, il nécessite une régularisation de la loi cohésive par pénalisation des conditions de contact et d'adhérence, apparemment source d'instabilités de la réponse globale de la structure. En outre, il conduit à des oscillations spatiales des forces cohésives qui peuvent s'avérer rédhibitoires, du fait de leur collocation aux points de Gauss des éléments.

Modélisations : `PLAN_JOINT, AXIS_JOINT, 3D_JOINT`
Documentations : [R3.06.09] et [U3.13.14]

L'élément à discontinuité (ED) incluse discrétise à la fois le saut de déplacement (constant par morceau) et le champ de déplacement volumique environnant ($Q1$). Il remplit son objectif, à savoir la prise en compte sans régularisation des conditions d'adhérence. En revanche, contrairement à ce qu'on espérait, la position géométrique de la fissure n'est pas laissée libre, sauf à s'écarter d'une formulation énergétique. En outre, sa limitation à un comportement élastique en quasi-statique et l'ordre de discrétisation insuffisant le rendent limité dans la pratique.

Modélisations : `PLAN_ELDI, AXIS_ELDI`
Documentations : [R7.02.14] et [U3.13.14]

Plus récemment, le développement d'une troisième famille d'éléments finis a permis de concilier les avantages des deux précédentes. Disposé lui aussi le long des fissures potentielles, **l'élément mixte d'interface (EI)** fournit une discrétisation du champ de déplacement ($P2$), des forces cohésives ($P1$) et du saut de déplacement ($P1$ -discontinu ou $P2$ -discontinu). De la sorte :

- il autorise le traitement des conditions d'adhérence sans aucune régularisation, d'où des réponses structurales plus régulières ;
- il respecte la condition LBB qui garantit une convergence satisfaisante avec le raffinement de maillage, avec le même taux que pour les éléments de volume (convergence quadratique) ;
- il est compatible avec toute loi de comportement volumique.
- en revanche il n'est compatible qu'avec un maillage quadratique

Pour les éléments d'interface, on distingue une modélisation standard `*_INTERFACE` et une modélisation sous-intégrée `*_INTERFACE_S`. Dans la première, le saut de déplacement est modélisé par un champ $P2$ -discontinu, tandis que dans la seconde il est modélisé par un champ $P1$ -

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

discontinu. Chacune de ces modélisations présente des avantages et des inconvénients. La modélisation standard limite le risque d'apparition de pivots nuls dans la matrice tangente, mais ne vérifie pas strictement la condition LBB. Pour y remédier, on conseille de choisir un paramètre de pénalisation du lagrangien augmenté pas trop grand (ce choix apporte un peu de « souplesse » dans la modélisation). La modélisation sous-intégrée vérifie mieux la condition LBB, mais présente plus souvent des pivots nuls dans la matrice tangente. Ces pivots apparaissent en particulier lorsque l'interface se situe sur un plan de symétrie, ou entre un volume et un élément de structure (MEMBRANE, GRILLE ou BARRE).

Les premières applications confirment le potentiel de cet élément qui aura réussi la synthèse des deux précédentes familles. Son application en dynamique pour traiter les propagations brutales est prévue à court terme.

Modélisations : PLAN_INTERFACE, AXIS_INTERFACE, 3D_INTERFACE, PLAN_INTERFACE_S, AXIS_INTERFACE_S, 3D_INTERFACE_S
Documentations : [R3.06.13] et [U3.13.14]

Modèle élément finis	EJ	ED	EI
Problème mécanique	Minimum de l'énergie	Minimum de l'énergie	Minimum de l'énergie sous contraintes
Modélisation	2D plan, axis et 3D	2D plan, axis	2D plan, axis et 3D
Dimension éléments finis	Surfacique	Volumique	Surfacique
Type d'éléments	QUAD4, HEXA8, PENTA6	QUAD4	QUAD8, HEXA20, PENTA15
Forme de l'énergie	Régularisée	Non régularisée	Non régularisée
Matériau environnant	Linéaire ou Non Linéaire	Linéaire	Linéaire ou Non Linéaire
Interpolation	Déplacement P1, Saut P1	Déplacement P1, Saut P0	Déplacement P2, Contraintes P1, Saut P1 discontinu ou P2 discontinu
Influence des paramètres numériques	Grande influence du paramètre numérique	Pas de paramètre numérique	Faible influence du paramètre numérique
Type de calculs	Statiques et dynamiques	Statiques	Statiques et dynamiques (?)

Tableau 1 : synthèse des avantages et limites des modèles élément finis cités précédemment.

Dans le tableau 1 on retrouve les avantages des modèles EJ et ED dans le modèle EI sans en avoir les limites : loi non régularisée, comportement matériau linéaire ou non linéaire, saut $P1$ et paramètre numérique qui n'influence pas les résultats du calcul tandis que c'est le cas avec celui des EJ.

2.3 Lois cohésives

Les lois cohésives sont une relation entre le vecteur contrainte et le saut de déplacement en chaque point de gauss. Toutes ces lois sont non-linéaires et doivent être utilisées avec les opérateur non-linéaires STAT/DYNA_NON_LINE (par conséquent elles ne sont pas adaptées pour MECA_STATIQUE). Pour les modèles précédents les lois cohésives sont baptisées :

EJ : CZM_LIN_REG , CZM_EXP_REG
ED : CZM_EXP
EI : CZM_OUV_MIX , CZM_TAC_MIX, CZM_EXP_MIX CZM_FAT_MIX, CZM_TRA_MIX, CZM_LAB_MIX

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

On donne un exemple en mode d'ouverture (figure 2.3), de la relation entre la composante normale du vecteur contrainte σ_n et le saut normal δ_n . La variable interne seuil κ permet de gérer l'irréversibilité de la fissuration. Les paramètres mécaniques de la loi sont la contrainte critique d'amorçage σ_c et le taux de restitution d'énergie critique G_c . Le rapport de ces deux paramètres définit un saut critique $\delta_c = 2G_c/\sigma_c$ au delà duquel la contrainte s'annule.

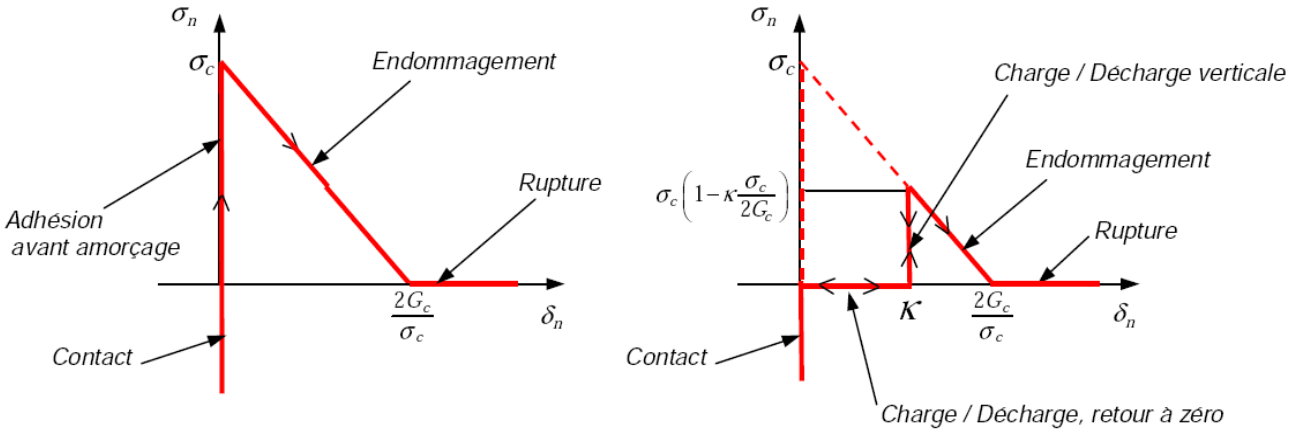


Figure 2.3 : Composante normale du vecteur contrainte en fonction du saut normal pour la loi CZM_OUV_MIX (seuil κ nul à gauche et positif à droite) .

Les principales différences entre les lois CZM* portent sur la phase d'amorçage et sur la forme de la loi post-pic. Les lois *REG prennent en compte la partie contact/adhésion de manière pénalisée tandis que les autres réalisent ces phases de manière parfaite (pente infinie à l'origine, voir figure 2.3). Notons que la régularisation des lois cohésives a une influence sur les résultats du calcul (voir la partie 15). Par ailleurs, les lois *EXP* (sauf CZM_EXP_MIX dont la loi a été tronquée volontairement) ont un comportement décroissant exponentiel qui tend asymptotiquement vers zéro avec l'ouverture des lèvres. Formellement, celles-ci ne permettent pas de définir une pointe de fissure, on leur préférera les autres lois à décroissance linéaire qui fournissent explicitement la position du fond de fissure (saut critique de rupture, voir figure 2.3).

D'une manière générale, lors de la phase de propagation de la fissure, la forme de la loi cohésive (linéaire ou exponentielle) a peu d'influence sur la fissuration, en revanche elle peut avoir un rôle sur la phase d'amorçage. Pour plus de détails sur ces lois voir [R7.02.11] .

2.4 Définition du repère local à la fissure

Pour l'intégration de la loi cohésive qui fait intervenir un saut de déplacement, fonction des déplacements nodaux exprimés dans le repère global, il est nécessaire de connaître un repère local $(n, t1, t2)$ à l'élément. Pour les EI et les EJ deux méthodes différentes ont été retenues, chacune ayant ses avantages mais aussi ses limitations.

Pour les EI : l'utilisateur fournit l'information nécessaire pour définir le repère local (en 2D, un angle de rotation suffit, en 3D il faut connaître les trois angles nautiques). C'est relativement simple pour une fissure rectiligne ou plane. Cela devient beaucoup plus complexe lorsque ce n'est plus le cas. En revanche cette méthode permet d'imposer un unique repère local à tous les éléments d'une fissure plane donnée et autorise ainsi une visualisation cohérente des sauts de déplacement en mode II et III . Nous verrons que ce n'est pas le cas pour les EJ et ED .
Pour un groupe de maille d'interface FISS dans un plan donné, l'utilisateur renseigne les caractéristiques des EI de la manière suivante (angles nautiques (a, b, c) en degrés) :

```
ORIEN_FI=AFFE_CARA_ELEM(
                    MODELE=MO,
```


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

```
MASSIF=(
       _F(GROUP_MA='FISS',ANGL_REP=(a, b ,c)),
                 ));
```

puis CARA_ELEM = ORIEN_FI dans STAT_NON_LINE.

En 2D il est facile de connaître l'angle entre la normale du repère local n et le vecteur X du repère global. Cet angle est le premier angle nautique, les deux autres sont renseignés à zéro. En 3D c'est un peu plus difficile. Pour aider l'utilisateur on fournit quelques exemples d'angles nautiques avec la définition du repère local $(n, t1, t2)$ dans le repère global (X, Y, Z).

$$(0,0,0):(n,t1,t2)=(X,Y,Z)$$
$$(90,0,0):(n,t1,t2)=(Y,-X,Z)$$
$$(-90,0,-90):(n,t1,t2)=(-Y,-Z,X)$$
$$(0,-90,-90):(n,t1,t2)=(Z,X,Y)$$

Pour les EJ et les ED : Le calcul du repère local est réalisé (en dur) à partir des coordonnées géométriques de l'élément. Aucune information n'est donnée par l'utilisateur, la prise en compte de fissures non rectilignes ou non planes est donc immédiate. En contrepartie, pour une fissure plane, les éléments ont tous la même normale mais des vecteurs tangents $(t1, t2)$ différents d'un élément à l'autre. Cela n'a bien sûr aucune incidence sur la validité des calculs, par contre cela empêche une visualisation propre des composantes du saut en mode II ou III.

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

3 Conseils sur le maillage

Après quelques généralités sur les différents type d'éléments finis, on fournit des conseils sur la réalisation du maillage ainsi que sur ses caractéristiques afin de modéliser correctement la fissure cohésive dans différentes configurations.

3.1 Généralités

D'une manière générale, les modèles cohésifs permettent de représenter une fissure sur un trajet Γ donné *a priori*. Ces modèles rendent compte de l'évolution mais pas de la direction de la propagation. Il est donc nécessaire de réaliser un maillage compatible avec le trajet de fissure potentiel. Cela consiste à disposer des éléments cohésifs le long de ce trajet.

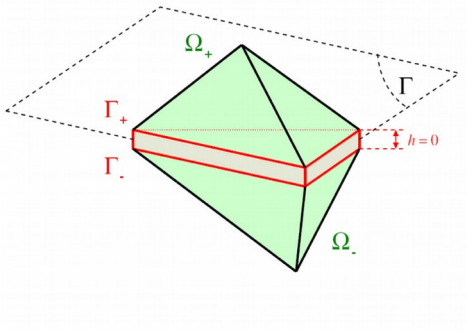


Figure 3.1-a : Schéma 3D d'un élément EI ou EJ inséré entre deux tétraèdres.

Pour les modèles surfaciques (EJ et EI), en 2D les lèvres de la fissure correspondent aux grands côté des éléments cohésifs (quadrangles dégénérés), en 3D les lèvres sont définies par les faces triangulaires (pour les pentaèdres dégénérés) ou rectangulaires (pour les hexaèdres dégénérés). Pour l'élément à discontinuité interne ED , la fissure passe par le centre de l'élément et est parallèle au deux côtés opposés. Les modèles EJ et ED ont une interpolation linéaire des déplacements, ils sont compatibles avec un maillage volumique linéaire $Q1$ ou $P1$. L' EI à une interpolation quadratique des déplacements et est compatible avec des mailles volumiques quadratiques $Q2$ ou $P2$.

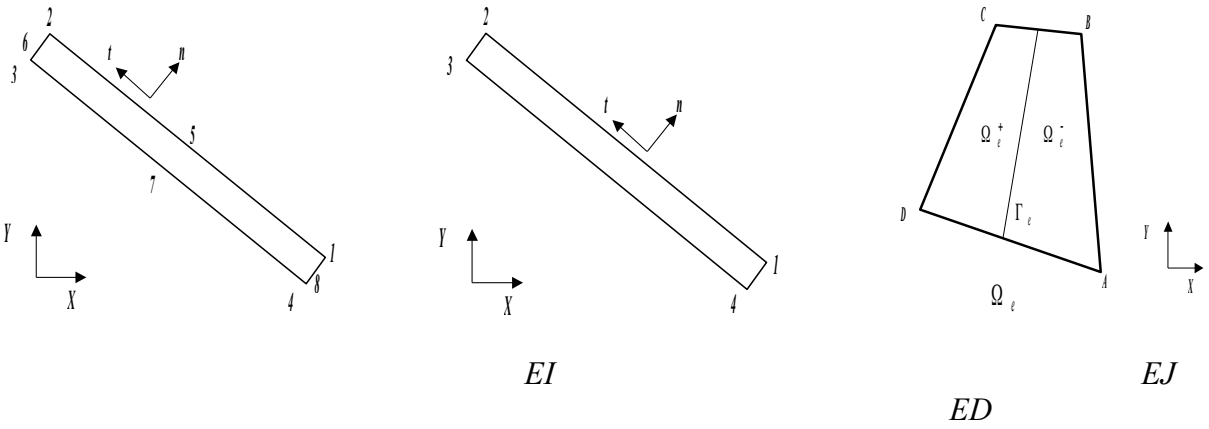


Figure 3.1-b : Schéma 2D des éléments cohésifs

3.2 Création du maillage

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

On donne dans cette partie les outils disponibles dans *Code_Aster* pour garantir la bonne réalisation d'un maillage avec une fissure cohésive puis on fournit des conseils pour les mailleurs GMSH et Salomé.

3.2.1 Outils *Code_Aster*

Les mailles des éléments cohésifs EI et EJ sont surfaciques, leur orientation est définie par un repère local, il est donc nécessaire que leur connectivité locale soit définie *a priori*. C'est également le cas des ED qui sont volumiques puisqu'ils sont traversés par une discontinuité surfacique.
Deux techniques sont disponibles dans *Code_Aster* pour réaliser cette tache. La première, une fois le maillage réalisé, consiste à vérifier qu'elles ont une numérotation locale *ad hoc* (orientation). La seconde consiste à réaliser le maillage des éléments cohésifs à partir de deux groupes de nœuds en regard (création automatique).

Remarques :

- *L'épaisseur des éléments EJ et EI n'a pas d'influence sur la solution. Cette dernière peut toutefois modifier les dimensions de la structure, l'utilisateur doit réaliser le maillage en conséquence. Lorsque cela est possible on conseille de réaliser des mailles d'épaisseur nulle.*
- *Pour le post-traitement en 2D une petite épaisseur peut toutefois faciliter la visualisation des variables aux points de gauss (ce problème ne se pose pas en 3D car on visualise une surface).*
- *En revanche, les ED ont nécessairement une épaisseur non nulle. On préconise de choisir une taille du même ordre de grandeur que celle des éléments non cohésifs environnants.*

Orientation
Le mot clé ORIE_FISSURE (commande MODI_MAILLAGE) permet de s'assurer de la bonne numérotation locale des mailles et de renuméroter correctement si nécessaire. Pour plus de détails sur l'utilisation de cette procédure voir [U4.23.04]. On conseille d'utiliser systématiquement cette commande qui fonctionne en 2D et en 3D (pourvu que l'épaisseur des mailles soit non nulle) et quel que soit le type d'éléments cohésifs (linéaire ou quadratique). On donne un exemple d'utilisation pour la couche cohésive 'JOINT' :

```
MA=MODI_MAILLAGE(reuse =MA,
                 MAILLAGE=MA, ORIE_FISSURE=_F(GROUP_MA='JOINT'),)
```

Création automatique
Une autre possibilité : CREA_FISS permet la création de mailles cohésives à partir de deux groupes de nœuds (en regard ou confondus deux à deux). Certaines précautions sont nécessaires pour l'utilisation de cette commande, il est donc conseillé de lire la documentation de CREA_MAILLAGE [U4.23.02]. Cette fonctionnalité est disponible en 2D (l'équivalent 3D est possible avec un python Salomé, voir 12 ci-dessous), en voici un exemple à partir des groupes de nœuds GN1 et GN2.

```
MA = CREA_MAILLAGE(
  MAILLAGE   = MA_INI,
  CREA_FISS = (
    _F(NOM='JOINT', GROUP_NO_1='GN1', GROUP_NO_2='GN2', PREF_MAILLE='MJ'),
    ))
```

Remarques :

- *Une fois la commande CREA_FISS réalisée il n'est pas nécessaire de passer la moulinette ORIE_FISSURE si on réalise un maillage linéaire .*
- *Si l'on souhaite utiliser cette procédure avec les éléments d'interface (donc maillage quadratique), il faut dans un premier temps réaliser la procédure précédente sur un maillage linéaire, puis passer en quadratique (LINE_QUAD/TOUT='OUI') et enfin repasser la moulinette ORIE_FISSURE sur la couche cohésive car la commande LINE_QUAD a peut être modifié la numération locale.*

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

Enfin, des outils sont en cours de développement pour faciliter la création de mailles cohésives. Notamment dans Salomé version 5 (voir 12).

3.2.2 Conseils GMSH

Avec le mailleur libre GMSH, il est assez facile de construire une couche d'éléments finis cohésifs destinés à être des éléments de joint ou d'interface quelle que soit la nature du maillage volumique. Il suffit d'utiliser la commande « `Extrude` » et de prendre quelques précautions.

En 2D : on extrude une ligne `n` d'un bord du domaine, d'une épaisseur `ep` (petite mais non nulle) dans la direction `-Y`, et on précise que l'on ne souhaite qu'une couche d'éléments avec le mot clé « `Layers`», enfin on ajoute « `Recombine` » pour que le mailleur crée des quadrangles (triangles par défaut).

```
ep=0.001
NbCouche = 1;
JOINT[]=Extrude Line {n, {0,-ep,0}}
            {
             Layers{NbCouche,1}; Recombine;
            };;
```

Ensuite on peut définir une « `Physical Line` » de la nouvelle ligne crée ou une « `Physical Surface` » de la surface crée de la façon suivante :

```
Physical Line(a)    = {JOINT[0]};
Physical Surface(b) = {JOINT[1]};
```

La première peut servir par exemple à l'application de conditions aux limites de symétrie, la seconde correspond à la couche d'éléments cohésifs. Si on souhaite inclure la fissure à l'intérieur d'un domaine, on procède comme précédemment, puis on extrude la nouvelle ligne `JOINT[0]` pour créer la seconde partie du volume.

En 3D : on procède de la même manière avec l'extrusion d'une surface frontière du domaine, le « `Recombine` » permet de créer des hexaèdres si la trace du maillage sur la surface extrudée comporte des quadrangles et des pentaèdres si cette dernière comporte des triangles.

Pour finir, les « physical » crées dans le .geo sont utilisé dans le fichier de commande .comm de façon habituelle :

`Physical Line(i)`	correspond à `'GMi'`	si le maillage est au format .msh
`Physical Surface(j)`	correspond à `'GMj'`	si le maillage est au format .msh
`Physical Line(k)`	correspond à `'G_1D_k'`	si le maillage est au format .med
`Physical Volume(l)`	correspond à `'G_3D_l'`	si le maillage est au format .med

Attention :

Il existe un bug dans GMSH 2.3 pour l'extrusion 3D avec l'algorithme de maillage 'Tetgen+Delaunay'. Cet algorithme est celui proposé par défaut, il est donc nécessaire de le modifier dans le menu Option/mesh/General et choisir 3D algorithm='Netgen'. Ce bug est répertorié sur le site web de GMSH et devrait être modifié dans les versions futures.

Remarques :

- *Les mailles créées par GMSH sont topologiquement correctes mais la numérotation locale des nœuds n'est pas nécessairement bien réalisée. Dans Code_Aster cette dernière doit être définie de manière spécifique afin de distinguer l'orientation des élément cohésifs. Après la lecture du maillage au format .msh ou .med il est donc nécessaire de passer par un `MODI_MAILLAGE` avec la commande `ORIE_FISSURE` qui réalise cette l'opération de renumérotation des éléments cohésifs de manière automatique (voir partie 10).*
- *Notons que les extrusions évoquées ci-dessus sont aussi facilement réalisables de façon interactive à l'aide de la souris. Une fois le .geo généré automatiquement, il faut toutefois*

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

ajouter le « Recombine » pour créer des quadrangles en 2D ou des hexaèdre/pentaèdres en 3D.

3.2.3 Conseils Salomé

Dans Salomé, il est possible d'extruder une couche d'éléments surfaciques d'un bord du domaine. Cela permet de créer des mailles cohésives dont on changera la connectivité si nécessaire à l'aide de la commande ORIE_FISSURE (voir partie 10).

Des travaux sont en cours pour réaliser le maillage d'éléments cohésifs entre deux volumes ou le long d'une couche surfacique dans un volume. La commande python DoubleNodeGroups[1] permet de dédoubler les nœuds d'un groupe de mailles surfaciques ou d'un groupe de nœuds donné (voir manuel de référence du module MESH). Le tissage des éléments cohésifs entre ces deux groupes de nœuds peut ensuite être réalisé en 2D dans le fichier de commande à l'aide du mot clé CREA_FISS (voir partie 10) ou à partir d'un script python[2] en 3D.

Remarque :

Quelle que soit la technique adoptée pour mailler le trajet de fissure potentiel il est nécessaire que celui-ci traverse de part et d'autre la structure. En effet, il n'existe pas d'éléments finis cohésifs destinés à modéliser une pointe de fissure (triangle par exemple) qui ne puisse se propager. Cela n'aurait d'ailleurs pas vraiment de sens puisque contraire à la logique de la modélisation d'un trajet potentiel.

3.2.4 Cas des configurations géométriques avancées

Lorsqu'on emploie les éléments finis d'interface (EI), la présence de deux familles de degrés de liberté, les déplacements et les forces cohésives (augmentées) complique la gestion des configurations géométriques avancées. Cette expression renvoie à des situations dans lesquelles plusieurs interfaces se croisent, débouchent l'une sur l'autre, débouchent sur une condition aux limites cinématiques ou encore présentent une frontière dans le volume et non pas sur un bord de la structure.

Des conseils sur la manière de mailler de telles configurations ont été donnés dans la note [bib9].On y observe que ce sont bien les nœuds qui portent les degrés de liberté correspondant aux forces cohésives qui nécessitent un soin particulier.

3.3 Modélisation de la zone cohésive

Une fissure CZM est définie par trois zones, la première correspond à une fissure libre de contrainte, la seconde, appelée zone cohésive, où les forces de cohésion sont non nulles et enfin une zone saine (ou zone de fissuration potentielle) où la contrainte d'amorçage n'a pas été atteinte (voir figure 3.3).
La zone cohésive permet de réaliser une transition progressive du matériau sain à une vraie fissure. La taille de cette dernière est liée à la longueur caractéristique G_c/σ_c du modèle cohésif, elle s'adapte pour supprimer la singularité des contraintes en pointe de fissure. Plus cette longueur est faible, plus la taille de la zone cohésive sera réduite. A la limite, si celle-ci tend vers zéro, la zone cohésive disparaît et on se retrouve dans le cas de la mécanique de la rupture classique avec la possibilité théorique d'avoir des contraintes infinies en pointe de fissure. Formellement, cela revient à prendre une contrainte à la rupture σ_c infinie.

On souligne ici l'importance tant du point de vue mécanique que numérique de simuler correctement cette zone cohésive. D'une manière générale et quel que soit le type de calcul, on conseille de mailler suffisamment cette zone de telle sorte qu'elle soit composée d'au moins quatre ou cinq éléments finis cohésifs. Cela permet de représenter finement l'évolution de la fissure sur un plan

1 Disponible uniquement en version 5 de Salomé
2 Une maquette en cours de validation est disponible chez l'auteur de cette documentation

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

mécanique mais permet également d'éviter des problèmes de convergence numérique liés à une discrétisation trop grossière de cette zone à fort gradient de contrainte.

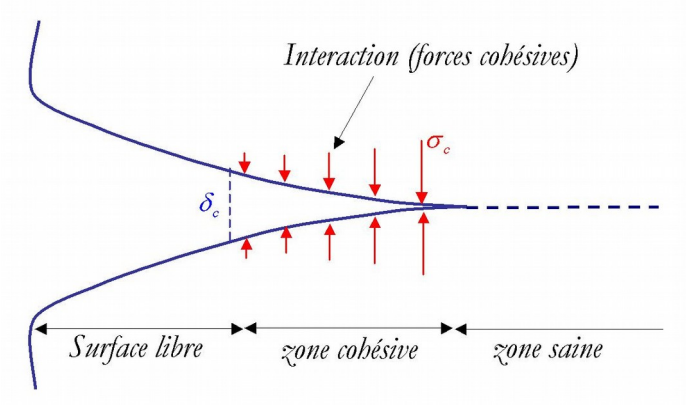


Figure 3.3 : Schéma de la fissure cohésive.

Dans la littérature [bib7] il existe une estimation de la taille de la zone cohésive S_{zc} pour une fissure dans un matériau infini élastique :

$$S_{zc} \simeq \frac{E}{1-\nu^2}\frac{G_c}{\sigma_c^2}$$

avec E module d'Young, ν coefficient de Poisson. Cette valeur est donnée à titre indicatif pour estimer la taille de zone cohésive, toutefois cette dernière peut varier avec différents paramètres du problème mécanique.

3.4 Définition d'une pré-fissure cohésive

Dans la plupart des cas, on dispose des éléments cohésifs dans un plan de fissuration ou dans le prolongement d'une fissure existante. Cette dernière est en général représentée par une surface libre. Toutefois, dans certains cas de figure, il peut être utile de disposer des mailles cohésives sur une fissure initiale. Ces dernières peuvent par exemple servir d'éléments de contact (petites déformations) ou servir à définir des fronts de fissuration initiaux plan ou semi-elliptiques par exemple.

Pour définir une fissure initiale, il est nécessaire de construire un maillage avec deux zones d'éléments cohésifs contiguës, l'une pour la pré-fissure et l'autre la fissure potentielle. L'idée consiste ensuite à imposer les variable internes de la pré-fissure de telles sorte que celles-ci correspondent à une zone libre de contrainte, et de ne pas modifier celles de la fissure potentielle (i.e. les imposer à 0). Pour ce faire, on utilise des commandes `CREA_CHAMP` pour construire un champ de variable interne aux point de gauss. Ce dernier est ensuite utilisé comme état initial en entrée dans `STAT_NON_LINE` avec la commande `ETAT_INIT`. Notons qu'il est nécessaire de définir l'ensemble des variables internes de la structure, y compris celles de la partie volumique. Dans le cas d'un matériau élastique cette opération est triviale, en non linéaire il n'est pas envisageable de les connaître facilement sauf à les imposer nulles ce qui n'est pas forcément réaliste.

Remarque :

Il n'est pas possible de définir une fissure initiale en imposant les paramètres matériaux σ_c et G_c nuls.

3.5 Multi-fissuration

Il est possible de modéliser plusieurs fissures avec tous les types d'éléments cohésifs pourvu que le maillage soit réalisé en conséquence. Avec l'élément ED on peut par exemple disposer des couches

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

d'éléments afin de prendre en compte un réseau de fissures parallèles (voir exemple [bib5]). Pour les élément EJ et EI le test de la plaque trouée (ssnp133 [V6.03.133]) permet de modéliser l'avancée simultanée de deux fissures.

Toutefois, avec ce type d'approche on déconseille de disposer des éléments cohésifs partout (i.e. sur chaque arrête du maillage). Outre la difficulté pour réaliser un tel maillage, de nombreux travaux dans littérature montrent que les résultats du trajet de fissuration sont fortement dépendants du maillage. De plus, pour les EJ, la régularisation de la loi cohésive peut également contribuer à la variabilité des résultats et jouer un rôle néfaste sur la qualité de la solution. Enfin cela pose la question non triviale des conditions à respecter à l'intersection de deux fissures. Aucun traitement particulier n'est réalisé pour les modèles présentés ici.

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

4 Conseils d'utilisation

Les modèles cohésifs couvrent un spectre assez large d'utilisation dans le domaine de la mécanique de la rupture. On commence les conseils d'utilisation sur des généralités concernant l'utilisation des paramètres et sur les chargements, valables quel que soit le type de simulation. On poursuit en donnant des conseils spécifiques pour les calculs en rupture fragile (matériau linéaire ou non linéaire), les calculs de fatigue et les calculs en dynamiques . Par la suite on souligne l'importance de s'appuyer sur des outils numériques pour améliorer la convergence des calculs. Enfin on termine sur des conseils pour le post-traitement des résultats.

4.1 Généralités

4.1.1 Paramètres mécaniques

Les paramètres mécaniques de la loi sont la contrainte critique d'amorçage σ_c et le taux de restitution d'énergie critique G_c . Sur la figure 2.3 à gauche, la contrainte σ_c correspond au pic de la loi, la densité G_c est reliée à l'aire sous le triangle. L'utilisateur définit un matériau et des valeurs (non nulles) de ces paramètres sous le mot clé RUPT_FRAG. Dans le cas d'un chargement thermique il est possible de faire dépendre ces derniers de la température. On utilise dans ce cas le mot clé RUPT_FRAG_FO, voir exemple dans le test SDNS105a.

Le rapport de ces deux paramètres définit un saut critique $\delta_c = 2G_c/\sigma_c$ au delà duquel la contrainte s'annule pour les lois CZM_LIN_REG, CZM_OUV_MIX, CZM_TAC_MIX . Pour les deux lois CZM_EXP_REG et CZM_EXP la contrainte tends asymptotiquement vers zéro. Pour la loi CZM_EXP_MIX, un saut critique est défini à $\delta_c = 3.2G_c/\sigma_c$ .Ce saut correspond à une erreur sur le taux de restitution d'énergie critique de 1 %. Pour la loi de fatigue CZM_FAT_MIX, la valeur du saut au delà de laquelle la contrainte reste nulle dépend du chargement. Pour la loi ductile CZM_TRA_MIX les paramètres de forme fournis par l'utilisateur influencent ce saut critique. Pour plus de détails voir la documentation [R7.02.11].

Remarque :

Lorsqu'on impose des conditions de symétrie sur les lèvres de la fissure il peut être nécessaire de diviser la valeur de G_c par deux, voir partie 16 .

4.1.2 Paramètres numériques

Éléments à discontinuité interne : Il n'y a pas de paramètre numérique à renseigner pour ce modèle.

Élément de joint : Ce modèle possède deux paramètres numériques destinés à pénaliser l'adhérence PENA_ADHERENCE et le contact PENA_CONTACT. On remplace les parties infiniment rigides (contact et adhérence) par des contributions élastiques fortement rigides (voir doc [R7.02.11]). Par défaut PENA_CONTACT vaut 1 et conduit à une pénalisation identique à celle de l'adhérence, c'est la valeur que l'on préconise. Le paramètre PENA_ADHERENCE correspond à un pourcentage de la longueur interne de la loi cohésive, il doit être non nul et inférieur à 1. Plus la valeur est faible plus la pente de pénalisation est importante.

Les résultats de calculs sont fortement dépendants de la valeur de ce paramètre. D'une manière générale, on observe :

- Des oscillations (snap back locaux) dans la réponse globale de la structure liées à l'ouverture successive des joints. Mais également des oscillations du profil de contrainte le long de la fissure d'autant plus marquée que la rigidité du joint est importante.
- Une influence importante du paramètre sur la qualité de la solution en contrainte. Pour des valeurs de PENA_ADHERENCE assez faibles l'erreur est raisonnable, pour des valeurs plus élevées cela peu conduire à des écarts très importants.

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

- La convergence de l'algorithme de Newton global est d'autant plus difficile que la pénalisation est importante.

On constate donc des effets antagonistes du paramètre de pénalisation qui impliquent de faire un compromis entre convergence du calcul et précision de la solution. Il est donc difficile de préconiser une règle générale fournissant la bonne valeur de ce paramètre. On conseille donc avant de réaliser une simulation numérique d'effectuer un test de sensibilité à ce paramètre. Dans tous les cas, la rigidité initiale des éléments de joint doit être plus importante que la rigidité des éléments volumiques voisins.

Élément d'interface : Ce modèle possède un paramètre numérique PENA_LAGR qui correspond à la pénalisation du lagrangien augmenté (voir [R3.06.13]). Les résultats de calcul ne dépendent pas de la valeur de ce paramètre. On conseille de prendre la valeur par défaut (100). Cette valeur doit être strictement supérieure à 1 afin d'assurer l'unicité de la solution lors de l'intégration de la loi cohésive.
Par ailleurs les lois CZM_OUV_MIX et CZM_TRA_MIX destinées à l'ouverture en mode I uniquement possèdent un paramètre baptisé RIGI_GLIS qui permet d'imposer une rigidité non nulle en glissement et éviter ainsi les problèmes de pivot nul lors de la rupture des éléments. On conseille d'utiliser la valeur par défaut de 10. La solution n'est pas affectée par ce paramètre pourvu que les modes de glissement ne soient pas activés.

4.1.3 Chargements et conditions aux limites

Tous les types de chargements mécaniques et thermiques peuvent être *a priori* appliqués sur une structure comportant des éléments cohésifs : déplacements imposés, efforts surfaciques, forces volumiques, champs de température issus d'un calcul thermique préalable.

Pour réaliser un gain en temps calcul, compte tenu des symétries du problème, il est souvent utile de simuler une demi-structure. Des conditions de symétrie sur les lèvres de la fissure cohésive (i.e. les faces des éléments cohésifs), conduisant à une ouverture en mode I pur, sont envisageables de deux façon différentes :

- La première consiste à bloquer les déplacements normaux des faces des éléments cohésifs qui ne sont pas reliés au volume. Ces dernières coïncident alors avec l'axe de symétrie de la structure. Dans ce cas, une précaution est à prendre sur la valeur de la densité d'énergie de surface critique entrée par l'utilisateur. Puisqu'on ne modélise qu'une lèvre de fissure, il faut, pour une valeur matériau donné de G_c , renseigner G_C= $G_c/2$ dans le fichier de commande. Autrement dit, cela coûte deux fois moins d'énergie d'ouvrir une demi-fissure.
- La seconde consiste à faire ouvrir la fissure entièrement. Sur les lèvres de la fissure on impose des déplacements symétriques vis-à-vis d'un axe passant entre celles-ci (commande AFFE_CHAR_MECA/LIAISON_GROUP). Ce type de modélisation permet de prendre en compte la fissure dans son ensemble et ne nécessite pas de diviser le G_c par deux. Par contre cela pose un problème sur le bilan énergétique puisqu'on modélise une demi-structure (donc une énergie volumique deux fois moindre) avec une fissure complète. Cette méthode est donc à utiliser avec beaucoup de précaution, on conseille plutôt d'adopter la précédente.

4.1.4 Critère de convergence

Le critère de convergence de l'algorithme de Newton RESI_REFE_RELA permet de construire une force nodale de référence, qui sert à estimer, la nullité (approchée) du résidu (voir [R5.03.01]). Celui-ci permet d'améliorer la convergence dans le cas où les résidus sont hétérogènes, c'est-à-dire lorsque l'on traite des degrés de liberté de nature différentes. C'est le cas du modèle mixte EI pour lequel on recommande l'utilisation de ce critère.

Avec les modèles cohésifs, la force est définie à partir d'une contrainte et de son dual un déplacement. Pour utiliser ce critère on renseigne donc une contrainte et un déplacement de référence : SIGM_REFE et DEPL_REFE. Pour choisir ces valeurs, on propose celles issues de la loi cohésive : la contrainte d'amorçage σ_c pour la première, et la longueur caractéristique G_c/σ_c pour le second. Avec les modèles EJ et ED il n'est pas possible (et *a priori* pas utile) d'utiliser ce type de résidu.

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

4.1.5 Solveur linéaire

La méthode de résolution du système linéaire global MUMPS est indispensable pour les éléments mixtes ayant des degrés de liberté de type Lagrange. Il est dont nécessaire de l'utiliser avec les EI. Son intérêt principal réside dans sa capacité à pivoter lignes et/ou colonnes de la matrice lors de la factorisation en cas de pivot petit. Les autres modèles éléments finis EJ et ED n'ont pas cette restriction, ils peuvent être utilisés avec la méthode par défaut (multifrontale).

Par ailleurs, la matrice tangente des lois cohésive est symétrique, l'utilisation du mot clé SYME='OUI' destiné à symétriser cette dernière n'est pas nécessaire.

4.2 Une interface linéaire pour valider sa mise en données

De la même manière qu'on recommande de réaliser un calcul élastique linéaire en préalable à un calcul plastique, il est préférable de débuter un calcul de fissuration en employant une relation de comportement cohésive **linéaire** . Le modèle CZM_ELAS_MIX a été introduit dans cette optique, voir [U4.51.11] . Cela permet de valider sa mise en données (orientation des interfaces, numérotation des degrés de liberté, affectation des caractéristiques et des chargements, etc.) dans un cadre suffisamment simple pour que l'intuition mécanique (ou l'échec du calcul) suffise à détecter d'éventuels résultats aberrants. Attention, toutefois : cette relation de comportement n'est disponible qu'avec les éléments d'interface (EI).

Ce modèle d'interface est élastique, avec une raideur dans la direction normale et une autre dans les deux directions tangentes. Elles sont définies via DEFI_MATERIAU [U4.43.01] sous le mot-clé facteur CZM_ELAS. Il est également possible de considérer des comportements limites via des conditions d'adhérence normale et/ou tangentielle parfaites, toujours sous le même mot clé facteur (dans ce cas, les raideurs ne jouent plus aucun rôle puisque le saut de déplacement dans la direction normale ou tangente est forcé à zéro). Typiquement, avec des conditions d'adhérence parfaite (normale et tangentielle), la réponse de la structure doit être très proche de celle qu'elle aurait sans interface (très proche mais pas nécessairement égale du fait de la formulation mixte utilisée avec l'élément EI).

Enfin, une dernière condition d'adhérence est proposée : il s'agit de conditions de contact en fermeture, assorties d'une raideur élastique en ouverture. Le calcul devient plus complexe car le problème n'est plus linéaire. Mais cela permet de s'assurer que les zones où l'interface s'ouvre et celles où elle reste fermée sont bien conformes à l'intuition.

4.3 Calculs en rupture fragile

Modèles possibles : EJ, EI, **Modèle conseillé** : EI

Les modèles cohésifs sont adaptés pour modéliser la rupture fragile dans un matériau linéaire (tous) ou non linéaire (EJ et EI seulement). On donne ici quelques conseils propres à ce type de simulations. Des outils numériques permettent de garantir une bonne robustesse des algorithmes de résolution, on fournit également des conseils sur leur utilisation.

4.3.1 Matériau linéaire

Pour un matériau élastique linéaire, il n'y pas de précautions de mise en œuvre spécifiques autres que les généralités précédentes (partie 15) et celle sur le maillage (partie 9). Pour la robustesse des calculs, l'utilisation du pilotage du chargement est primordiale. On consacre ci-dessous une partie aux conseils d'utilisation de cette méthode.

4.3.2 Matériau non linéaire

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

Les modèles cohésifs EJ et EI peuvent interagir avec un matériau non linéaire environnant *a priori* quel qu'il soit. Les simulations envisageables restent dans le domaine de la rupture fragile en présence de plasticité, celle-ci étant confinée au voisinage de la pointe de fissure. Aucun traitement numérique spécifique n'est nécessaire pour ce type de calcul, si ce n'est les précautions habituelles sur la finesse du maillage (voir partie 9) afin de capter la zone cohésive mais également la bande plastique le long des lèvres de la fissure. La convergence de l'algorithme de Newton-Raphson est ralentie (plus d'itérations de Newton à pas de chargement donné) et le temps calcul est augmenté mais cela reste tout a fait raisonnable et n'entrave pas la robustesse des modèles cohésifs dans ce domaine d'application.

D'un point de vue mécanique, la plastification du matériau ainsi que la fissuration sont deux phénomènes qui entrent en interaction. Ainsi, une attention particulière devra être apportée dans le choix du rapport de la contrainte d'amorçage de la loi cohésive et de la limite d'élasticité du matériau.

Suivant les valeurs de ces paramètres, on pourra se trouver soit dans le cas de figure où la fissuration intervient sans développement de la plasticité, soit dans le cas contraire où la contrainte dans le matériau atteint le seuil plastique et évolue sans dépasser la contrainte d'amorçage, ou enfin le cas de figure où plasticité et fissuration évoluent conjointement. Dans ce dernier cas, on observe une zone plastique qui s'initie en pointe de fissure et qui évolue avec la propagation de celle-ci.

Par ailleurs, quelques calculs avec des matériaux viscoplastiques ont été réalisés à titre illustratif, le faible retour d'expérience dans ce domaine ne nous permet pas de fournir de conseils spécifiques.

4.3.3 Outils numériques

Recherche linéaire
Il est possible d'utiliser la recherche linéaire avec les modèles cohésifs. Cela peut ponctuellement améliorer la convergence des calculs. Cette méthode n'est toutefois pas la mieux adaptée pour franchir les instabilités liés à la propagation brutale de fissure en quasi-statique. Dans ce cas de figure, on lui préférera le pilotage du chargement.

Pilotage du chargement
Le pilotage du chargement, disponible pour tous les modèles cohésifs, permet d'assurer la convergence de l'algorithme de Newton-Raphson en quasi-statique malgré une instabilité dans la réponse globale de la structure (propagation brutale de fissure par exemple). Cette méthode est valable pour les lois adoucissantes qui peuvent conduire à ce type de réponses et en particulier pour les lois cohésives. L'idée consiste à autoriser le relâchement du chargement pour restituer une partie de l'énergie potentielle et faire avancer la fissuration progressivement en contrôlant localement l'endommagement des zones cohésives.

L'utilisateur fournit la direction du chargement (déplacement ou effort surfacique) à partir d'un AFFE_CHAR_MECA, celle-ci est multipliée par le paramètre de pilotage scalaire η, nouvelle inconnue du problème qui donne son intensité. Ce dernier paramètre remplace la fonction multiplicative du chargement FONC_MULT. Lors de la résolution du problème mécanique, on ajoute une équation supplémentaire (contrôle du pilotage) qui ferme le système. Pour plus de détails sur cette technique, le lecteur peut se reporter à la documentation générale [R5.03.80] et aux documentations spécifiques au pilotage pour les modèles cohésifs [R7.02.11] et [R7.02.14].

Cet outil numérique est particulièrement important pour assurer la convergence de l'algorithme. Il est souvent nécessaire, y compris lorsque la réponse globale n'admet pas de retour arrière. En effet, dans certaines configurations, des retours arrières locaux, liés à une ouverture brutale au niveau élémentaire, peuvent intervenir. Toutes les lois cohésives peuvent être pilotées par prédiction élastique (PRED_ELAS), le rapport du pas de temps Δt sur le coefficient COEF_MULT permet de gérer l'évolution de la fissuration. Au niveau de la loi de comportement cela revient à contrôler la sortie du seuil. Plus ce rapport est élevé plus on souhaite passer l'instabilité rapidement et donc plus η sera élevé.
Dans la commande STAT_NON_LINE, on choisit de piloter la charge TRACTION (renseignée classiquement) de la façon suivante :

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

```
  EXCIT      = _F(CHARGE = TRACTION , TYPE_CHARGE = ' FIXE_PILO '),
```

On ajoute ensuite les données propres au pilotage :

```
  PILOTAGE   = _F(
                  SELECTION = 'RESIDU',
                  TYPE = 'PRED_ELAS',
                  GROUP_MA = 'JOINT',
                  COEF_MULT = c_mult_pilo,
                  ),
```

Remarques et conseils :

- *On conseille de définir la direction du chargement piloté par un vecteur unitaire. Ainsi, le paramètre de pilotage η correspondra exactement à son intensité.*
- *On conseille à l'utilisateur de commencer par choisir un rapport Δt sur COEF_MULT égal à 0.05. De la sorte, on cherche à endommager de 5% le point de Gauss le plus sollicité.*
- *Si l'algorithme ne converge pas malgré le pilotage, on conseille de réduire le rapport précédent : sous découpage automatique ou manuel du pas et/ou augmentation de COEF_MULT. Plus le rapport sera faible, plus la fissuration sera lente et inversement.*
- *Il est parfois nécessaire de rompre la symétrie du problème pour éviter que le pilotage oscille entre deux valeurs de η si celles-ci fournissent un état de fissuration identique. Pour cela, on conseille de réaliser un premier calcul non linéaire avec un pas de chargement non piloté très faible. On reste ainsi dans le domaine linéaire de la réponse et le calcul converge rapidement.*
- *Une autre manière d'éviter les oscillations de la valeur du pilotage consiste à renseigner un ETA_PILO_R_MIN égal à zéro afin de n'autoriser que des valeurs positives du paramètre de pilotage.*
- *Il est nécessaire de piloter finement (faible sortie du critère) les premiers pas de chargement piloté afin de capter correctement la valeur pic de la réponse globale.*
- *Si le pilotage est utilisé dans un cas de figure où l'on pourrait s'en affranchir, la solution n'est pas modifiée. La seule différence est que l'on ne maitrise pas directement l'intensité de la charge.*
- *Il n'est pas possible de piloter des chargements thermiques.*

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

4.4 Calculs en rupture ductile

Modèles possibles : EI

La loi CZM_TRA_MIX a été introduite en version 10.4 pour chercher à modéliser la rupture ductile. L'utilisation de celle-ci doit être circonscrite à un cadre de recherche amont. En effet, elle est en cours de validation et sera certainement amenée à évoluer.

La loi pour la rupture ductile s'inspire des lois classiquement proposées dans la littérature. En plus des paramètres de contrainte critique et de densité d'énergie critique, on ajoute deux paramètres liés à la forme de la loi (COEF_EXTR et COEF_PLAS). Cette dernière est choisie trapézoïdale. Des calculs de sensibilité aux paramètres ainsi que des tests sur éprouvettes CT (SSNP151) et AE (SSNA120) ont été réalisés. Les premières conclusions montrent que ce modèle permet de reproduire correctement la réponse globale et la cinétique de propagation observées avec un modèle de type Gurson (Rousselier) sur une CT 2D (SSNP151a). On note toutefois que le recalage des paramètres de la loi pour s'approcher du calcul fin de l'approche locale reste difficile. En revanche le gain en temps calcul et en robustesse est significatif. Lorsqu'on passe sur une CT 3D (SSNP151b) les résultats sont moins concluants à partir d'un certain niveau de chargement. Il semble que cela soit en partie dû à la mauvaise prise en compte des effets volumiques dans le modèle cohésif et notamment l'effet de confinement. Un travail est en cours pour essayer de prendre en compte l'influence de la triaxialité. D'autres limites sont également à souligner : comme pour les autres modèles cohésifs, le trajet de fissure doit être connu *a priori* . La fissuration ne peut se faire qu'en mode I (il est toutefois envisageable d'introduire les modes de glissement si cela s'avère judicieux). Enfin, les grandes ouvertures de fissures ne sont pas prises en compte, on peut toutefois utiliser des grandes déformations dans le matériau environnant.

Aucun traitement numérique spécifique n'est nécessaire pour ce type de calcul, si ce n'est les précautions habituelles sur la finesse du maillage (voir partie 9) afin de capter la zone cohésive mais également la bande plastique le long des lèvres de la fissure. La convergence de l'algorithme de Newton Raphson est ralentie (plus d'itérations de Newton à pas de chargement donné) et le temps calcul est augmenté mais cela reste tout a fait raisonnable et n'entrave pas la robustesse des modèles cohésifs dans ce domaine d'application.

D'un point de vue mécanique, la plastification du matériau ainsi que la fissuration sont deux phénomènes qui entrent en interaction. Ainsi, une attention particulière devra être apportée dans le choix du rapport de la contrainte d'amorçage de la loi cohésive et de la limite d'élasticité du matériau. Suivant les valeurs de ces paramètres, on pourra se trouver soit dans le cas de figure où la fissuration intervient sans développement de la plasticité, soit dans le cas contraire où la contrainte dans le matériau atteint le seuil plastique et évolue sans dépasser la contrainte d'amorçage, ou enfin le cas de figure où plasticité et fissuration évoluent conjointement. Dans ce dernier cas, on observe une zone plastique qui s'initie en pointe de fissure et qui évolue avec la propagation de celle-ci.

4.4.1 Outils numériques

Recherche linéaire
Il est possible d'utiliser la recherche linéaire. Cela peut ponctuellement améliorer la convergence des calculs.

Pilotage du chargement
On considère que les fissures se propagent de façon progressive en mode de rupture ductile. Le pilotage du chargement n'est donc pas développé pour ce type de loi.

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

4.5 Calculs en fatigue

Modèle possible : EI

La loi disponible pour traiter du phénomène de fatigue en mode I est CZM_FAT_MIX (voir doc [R7.02.11]) utilisable avec les éléments d'interface. On peut réaliser des calculs en 2D ou 3D quel que soit le matériau et pour tous types de chargements mécaniques (d'amplitude constante ou variable), ou chargements thermiques. On pourra également se reporter aux tests SSNP118JKL et SSNP139B et à la partie 25.

Pour rendre compte de la fatigue avec cette loi, il est nécessaire de parcourir tous les cycles du chargement. On comprend aisément qu'il n'est pas envisageable de simuler un grand nombre de cycles. Néanmoins, les premiers résultats mettent en évidence un certain nombre de phénomènes observés en fatigue (voir compte rendu [bib3]). Pour un matériau élastique :

- On arrive à reproduire des lois d'évolution de type Paris dont on identifie les coefficients *a posteriori*,
- On note la faible influence d'une surcharge sur l'avancée,
- On observe l'absence d'effet de séquence à amplitude variable,
- On simule des évolutions de front de fissure 3D réalistes.

En présence de plasticité plusieurs effets observés expérimentalement sont reproduits :

- On met en évidence l'effet de surcharge qui conduit au ralentissement de la propagation dû à la compression induite par l'augmentation de la zone plastique,
- le ralentissement est d'autant plus prononcé que la surcharge est importante,
- l'augmentation du rapport de charge contribue à réduire le retard à rapport de surcharge et amplitude de charge fixés.

Par ailleurs des travaux sur le saut de cycles sont envisagés. Il permettront, par extrapolation, de simuler un grand nombre de cycles.

Conseils sur la discrétisation du chargement :

- *Lors de la discrétisation du chargement cyclique quelques précautions sont à prendre pour optimiser le temps calcul. Le risque est de réaliser à chaque cycle un redécoupage automatique du pas qui peut devenir très couteux en CPU. Le retour d'expérience nous permet de conseiller à l'utilisateur de réaliser environ 4 incréments du chargement par cycle lorsque le matériau est élastique et de doubler cette valeur lorsque celui-ci à un comportement non linéaire. Cette règle n'est pas nécessairement valable dans toutes les situations et devra être adaptée au cas par cas pour optimiser le temps passé dans chaque cycle.*

Conseil sur la définition de la fonction cyclique :

Pour la construction de la fonction multiplicative du chargement on peut utiliser DEFI_FONCTION ou du python dans le fichier de commande. On propose ci-dessous un exemple de fonction en dent de scie à d'amplitude variable. Les instants impairs sont les fins de monté en charge (amplitude max) et les instants pairs sont les fins de décharge (amplitude min). De la sorte, on termine le i ème cycle à l'instant $2i$. Dans l'exemple ci dessous le chargement varie
De val_inf à val_sup au cours des cycles 1 à nb_cy_1,
De val_inf à val_surch au cours des cycles nb_cy_1+1 à nb_cy_2,
De val_inf à val_sup au cours des cycles nb_cy_2+1 et nb_cy_3

Exemple de fonction cyclique f d'amplitude variable :

```
f = []
for i in xrange(nb_cy_1) :
  f.extend((2*i,val_inf,2*i+1,val_sup))

for i in range(nb_cy_1,nb_cy_2) :
  f.extend((2*i,val_inf,2*i+1,val_surch))

for i in range(nb_cy_2,nb_cy_3) :
  f.extend((2*i,val_inf,2*i+1,val_sup))
```


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

La définition du concept fonction dans *Code_Aster* est ensuite réalisé classiquement :

```
FCT_FAT = DEFI_FONCTION(
                        NOM_PARA = 'INST',
                        PROL_GAUCHE='LINEAIRE',
                           VALE     = f,
                        )
```

Conseil sur le stockage des données

Pour éviter un stockage des bases trop importantes on conseille de ne garder que les valeurs des champs au sommet de chaque cycle à l'aide du mot clé ARCHIVAGE de STAT_NON_LINE. Pour ce faire, il est bien sûr nécessaire de réaliser une discrétisation du chargement qui passe par les sommets.

4.6 Calculs en dynamique

Modèles possibles : EJ , EI (en cours de validation), **Modèle conseillé** : EJ

Le fait de prendre en compte les aspects dynamiques introduit une complexité supplémentaire dans la modélisation. Outre les précautions d'usage en quasi-statique (raffinement de maillage suffisant pour discrétiser la zone cohésive, relativement petite pour les caractéristiques des aciers (de l'ordre de 1 mm)), le régime dynamique impose des règles supplémentaires. On propose les quelques conseils pratiques suivants reposant sur le retour d'expérience. On pourra également se reporter au compte rendu [bib1].

- Le schéma d'intégration tout d'abord : le schéma de Newmark dit « d'accélération moyenne ($d=1/2$, $a=1/4$), inconditionnellement stable et non dissipatif en milieu linéaire, semble le mieux adapté (pas de dissipation numérique décelée en présence de zones cohésives). Très peu de tests ont été expérimentés avec d'autres schémas (explicites, Wilson…) mais ont conduit à des problèmes de convergence. Une thèse est en cours (D. Doyen) pour optimiser des schémas en présence de chocs.
- S'agissant de la discrétisation temporelle, tout dépend des phénomènes que l'on cherche à simuler. Pour cerner finement les réflexions des ondes sur la zone cohésive (entraînant une perturbation de la cinématique de la fissure), il faut (toujours pour les caractéristiques habituelles des aciers) considérer des pas de temps inférieurs ou égaux à $10^{-8} s$. Si un suivi plus grossier de la cinématique est suffisant et que le phénomène important est l'amorçage et l'arrêt, alors des pas de l'ordre de $10^{-7} \leq \Delta t \leq 10^{-5} s$ suffisent. Au dessus, des problèmes de convergence apparaissent ($\Delta t \approx 10^{-6} s$ semble être un bon compromis).
- Une attention toute particulière doit être portée sur le passage d'une phase stable à une phase instable (plus précisément une phase où la zone cohésive s'étend de façon progressive, puis où elle se propage de manière brutale). Dans ce cas, il faut adapter les pas de temps pour capter l'instant où la zone cohésive accélère, sous peine de gommer complètement « l'effet dynamique ». L'utilisation du pilotage du chargement en quasi statique, associé à l'observation du snap back de la réponse globale peut fournir des informations sur cet instant critique.
- Le petit paramètre PENA_ADHERENCE représentant la raideur du joint avant la décohésion (paramètre de régularisation) doit être pris inférieur à 10^{-4} . En effet, en dessous de cette valeur les résultats ne sont plus sensibles à ce paramètre.

D'une manière générale, le retour d'expérience est limité dans ce domaine d'application. L'utilisation de ces modèles en dynamique reste, pour le moment, destinée à des spécialistes.

4.7 Post-traitement de la fissuration et commande POST_CZM_FISS

Visualisation de la fissuration pour les EJ et EI

Pour visualiser une fissure cohésive, plusieurs solutions sont possibles :

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

- La variable interne au point de gauss (VARI_ELGA) $V3$ est un **indicateur de fissuration** qui permet de savoir si le point de gauss est sain ($V3=0$), endommagé ($V3=1$) ou rompu ($V3=2$). Voir l'exemple figure 4.5.
- La variable $V4$ varie continûment de 0 (sain) à 1 (rompu) ; elle correspond au **pourcentage d'énergie de surface dissipée** et peut être interprétée comme un indicateur d'endommagement . Voir l'exemple figure 4.5.
- On peut également visualiser les valeurs du **saut de déplacement** dans les trois modes de rupture I, II et III respectivement à partir des variables internes $V7$, $V8$ et $V9$.
- Un **indicateur de dissipation** $V2$ permet de savoir si le point de gauss dissipe de l'énergie ou est en décharge (les valeurs prises par cette variable interne dépendent de la loi cohésive, voir doc [R7.02.11]).
- Enfin on peut visualiser les composantes de la force cohésive dans les trois modes de rupture I , II et III respectivement à partir du vecteur contrainte au point de gauss SIEF_ELGA : SIGN , SITX et SITY . Le signe de SIGN permet de savoir si le point est en contact ou pas.

Attention, pour les EJ la visualisation des sauts tangentiels en mode II et III (respectivement $V8$ et $V9$) n'a pas de sens compte tenu du fait que les vecteurs tangentiels du repère local sont indépendants d'un élément à l'autre (voir à ce sujet la partie 7).

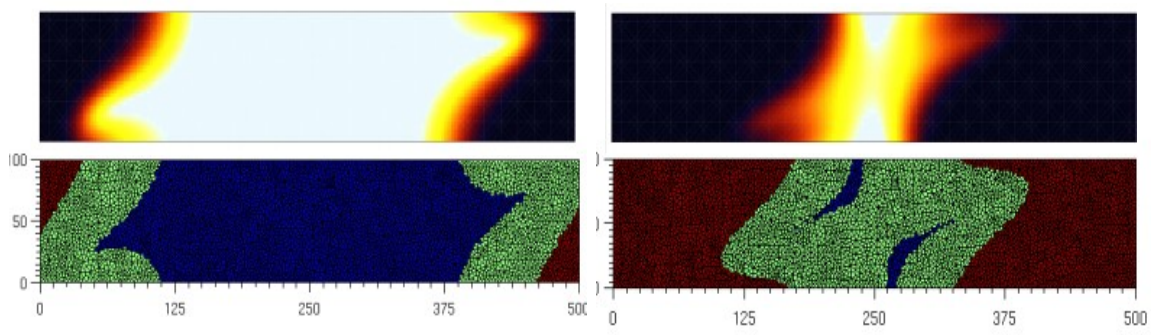


Figure 4.5 *:* Evolution de la rupture en mode mixte dans une poutre 3D, visualisation du plan de fissuration. Variable interne $V4$ en haut (1 en noir, 0 en blanc) et $V3$ en bas (0 en bleu, 1 en vert, 2 en rouge).

Visualisation de la fissuration pour les ED

Pour les ED les variables internes ne sont pas les même, le pourcentage d'énergie dissipée est stockée dans la $V5$ et l'indicateur de dissipation dans la variable $V4$. On stocke également les contraintes normale et tangentielle dans $V6$ et V7 ainsi que les sauts de déplacement normal et tangentiel dans $V1$ et $V2$.

POST_CZM_FISS : calcul de la longueur de fissure cohésives en 2D

Cette commande permet de connaître, à chaque instant, la taille de la fissure cohésive ainsi que la taille de la zone cohésive à partir des coordonnées d'un point POINT_ORIG et d'un vecteur directeur VECT_TANG donnés par l'utilisateur. Ce dernier doit également renseigner le résultat de calcul non-linéaire evol_noli ainsi que le groupe de mailles cohésives grma sur lequel il souhaite effectuer le post-traitement. La syntaxe est la suivante :

```
table_sdaster = POST_CZM_FISS(RESULTAT = evol_noli,
                              GROUP_MA = grma,
                              POINT_ORIG = (R,R),
                              VECT_TANG =(R,R))
```


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

Cette commande est utilisable pour les modélisations PLAN_JOINT, PLAN_INTERFACE, AXIS_JOINT et AXIS_INTERFACE et les lois cohésives CZM_OUV_MIX, CZM_TAC_MIX, CZM_LIN_REG, CZM_EXP_REG. Les longueurs de fissures sont évaluées à partir de la position géométrique des points de Gauss et de la variable interne V3 des lois cohésives qui permet de connaître l'état du point (sain, endommagé, cassé). On se reportera à la documentation U4 de POST_CZM_FISS pour plus d'informations. Un exemple d'utilisation est proposé dans les tests ssnp139a et ssnp139c.

Calcul de la longueur de fissure en 3D
Il est envisageable d'étendre la macro ci-dessus pour le 3D mais cela n'est pas prévu à court terme.

5 Mise en œuvre d'un calcul CZM

Les modèles cohésifs présentés ici ont fait l'objet d'un certain nombre de calculs dans différents problèmes mécaniques, le plus souvent sur des structures cibles (éprouvettes). Citons quelques exemples :

- Amorçage et propagation de fissure dans une poutre élastique ou élastoplastique en rupture fragile,
- propagation d'une fissure sous chargement cyclique dans une poutre élastoplastique pour modéliser la fatigue ainsi que l'effet de surcharge.
- Propagation d'une fissure 3D plane en rupture fragile ou fatigue, suivit du front de fissuration.
- Modélisation de la propagation dynamique de fissure dans un bimatériau.
- Fissuration et propagation brutale de deux fissures à travers une plaque trouée (calculs quasi-statiques avec pilotage).
- Arrachement d'une armature rigide.

Dans cette partie, on présente les différents tests s'appuyant sur les éléments cohésifs dans *Code_Aster* puis on commente l'un d'entre eux.

5.1 Liste des cas tests avec les modèles cohésifs

TEST ELEMENTAIRE, UN ELEMENT COHESIF ET UN ELEMENT ELASTIQUE

```
ssnp118a : EJ 2D PLAN,    LOI CZM_EXP_REG
ssnp118b : EJ 3D (HEXA),  LOI CZM_EXP_REG
ssnp118c : EJ 3D (PENTA), LOI CZM_EXP_REG
ssnp118d : EJ 2D PLAN,    LOI CZM_LIN_REG
ssnp118e : EJ 3D (HEXA),  LOI CZM_LIN_REG
ssnp118f : EJ 3D (PENTA), LOI CZM_LIN_REG
ssnp118g : EI 2D PLAN,    LOIS CZM_OUV_MIX ET CZM_TAC_MIX
ssnp118h : EI 3D (HEXA),  LOIS CZM_OUV_MIX ET CZM_TAC_MIX
ssnp118i : EI 3D (PENTA), LOIS CZM_OUV_MIX ET CZM_TAC_MIX
ssnp118j : EI 2D PLAN,    LOI CZM_FAT_MIX
ssnp118k : EI 3D (HEXA),  LOI CZM_FAT_MIX
ssnp118l : EI 3D (PENTA), LOI CZM_FAT_MIX
ssnp118m : EI 2D PLAN,    LOI CZM_TRA_MIX
ssnp118n : EI 3D (HEXA),  LOI CZM_TRA_MIX
ssnp118o : EI 3D (PENTA), LOI CZM_TRA_MIX
ssnp118q : EI 2D PLAN,    LOI CZM_EXP_MIX
ssnp118r : EI 3D (HEXA),  LOI CZM_EXP_MIX
ssnp118s : EI 3D (PENTA), LOI CZM_EXP_MIX
```

MISE EN OEUVRE DE LA LOI D'INTERFACE ELASTIQUE SUR UN CUBE

```
ss lv01 a  : E I 3 D (PENTA) , LOI CZM_ ELAS _ MIX
sslv01b  : EI 2D (TRIA),  LOI CZM_ELAS_MIX
```

ARRACHEMENT D'UNE ARMATURE RIGIDE

```
ssna115a : ED 2D AXIS, LOI CZM_EXP
ssna115b : EJ 2D AXIS, LOI CZM_LIN_REG
```


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

ssna115c : EI 2D AXIS, LOI CZM_TAC_MIX

EPROUVETTE AXISYMETRIQUE ENTAILLEE (AE) ELASTOPLASTIQUE
ssna120a : EI 2D AXIS, LOI CZM_TRA_MIX

FISSURATION D'UNE PLAQUE, COMPARAISON AVEC SOLUTION ANALYTIQUE
ssnp128a : ED 2D PLAN, LOI CZM_EXP

FISSURATION BRUTALE D'UNE PLAQUE TROUEE AVEC PILOTAGE DU CHARGEMENT
ssnp133a : EJ 2D PLAN, LOI CZM_EXP_REG
ssnp133b : ED 2D PLAN, LOI CZM_EXP
ssnp133c : EI 2D PLAN, LOI CZM_OUV_MIX

POUTRE Compact Tension (CT) 2D ELASTOPLASTIQUE
ssnp151a : EI 2D PLAN, LOI CZM_TRA_MIX

POUTRE Compact Tension (CT) 3D ELASTOPLASTIQUE
ssnp151b : EI 3D (HEXA), LOI CZM_TRA_MIX

POUTRE Double Cantilever Beam (DCB) 2D ELASTOPLASTIQUE, MONOTONE ET FATIGUE
ssnp139a : EI 2D PLAN, LOI CZM_OUV_MIX
ssnp139b : EI 2D PLAN, LOI CZM_FAT_MIX *(exemple commenté ci-dessous)*

POUTRE Double Cantilever Beam (DCB) 3D ELASTIQUE
ssnv199a : EJ 3D (HEXA), LOI CZM_EXP_REG
ssnv199b : EJ 3D (PENTA), LOI CZM_EXP_REG
ssnv199c : EJ 3D (HEXA), LOI CZM_LIN_REG
ssnv199d : EJ 3D (PENTA), LOI CZM_LIN_REG
ssnv199e : EI 3D (HEXA), LOI CZM_OUV_MIX
ssnv199f : EI 3D (PENTA), LOI CZM_OUV_MIX

PROPAGATION 2D DYNAMIQUE ET DEPENDANCE EN TEMPERATURE DE LA LOI COHESIVE
sdns105a.comm : EJ 2D PLAN, LOI CZM_EXP_REG

POUTRE ENTAILLEE EN FLEXION 3 POINTS
ssnp166a.comm : EI 2D PLAN, LOI CZM_EXP_MIX

5.2 Exemple commenté, poutre DCB en fatigue

On présente ici le test SSNP139, modélisation b, correspondant à la simulation de la propagation de fissure par fatigue dans une poutre Double Cantilever Beam (DCB) bidimensionnelle. On présente le schéma du problème mécanique sur la figure 5.2 puis le fichier de commande commenté en faisant référence aux différents conseils prodigués ci-dessus. On souligne en rouge les passages spécifiquement dédiés aux éléments cohésifs.

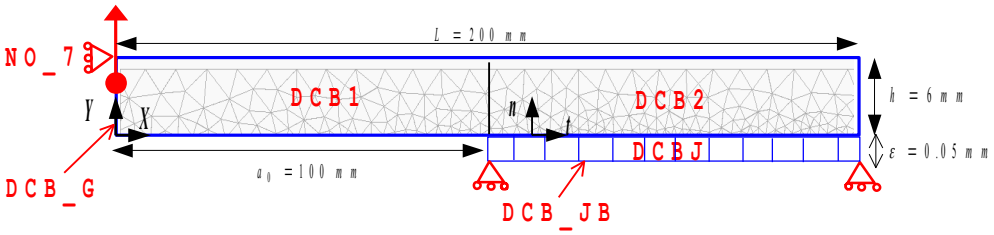


Figure 5.2 : Géométrie de la DCB et conditions aux limites.

```
#########################################################################
# POUTRE DCB 2D PLASTIQUE AVEC ELEMENTS D'INTERFACE EN FATIGUE : LOI CZM_FAT_MIX
```


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

```
#########################################################################################

DEBUT(CODE=_F( NOM = 'SSNP139B'))

# LOIS DE COMPORTEMENT DE LA DCB ET DE LA FISSURE COHESIVE (voir 6)
loi_vol='VMIS_ISOT_LINE'
loi_czm='CZM_FAT_MIX'

# NOMBRE DE CYCLES MAXIMUM
nb_cy_max= 101

# INSTANT FINAL (IMPAIRE ET INFERIEUR A 2*nb_cy_max-1)
tfin=5

# NOMBRES DE PAS (voir 21)
nbpas = 5*tfin

# ON ARCHIVE LES SOMMETS (voir 21)
nbarch = (tfin-1)/2

# VALEURS DE LA FONCTION DE CHARGEMENT
delta_U = 0.4
val_inf = 0.0
val_sup = val_inf + delta_U
val_surch = 1.2

# NUM DE CYCLE POUR CHANGEMENT AMPLITUDE VARIABLE
# pour faire une surcharge unique à l'instant 2*x+1 mettre : nb_cy_1 = x, nb_cy_2 = x+1 , nb_cy_3=... ,
# ci dessous, surcharge à l'instant 23.
nb_cy_1 = 11
nb_cy_2 = 12
nb_cy_3 = nb_cy_max

# PARAMETRES MECANIQUES DE LA LOI CZM (valeurs tests, voir 15)
tenac = 0.259/2 # on ne modélise qu'une demi fissure (division par deux du Gc matériau voir 15)
sigc = 35
dc = 2*tenac/sigc

# PARAMETRES NUMERIQUES DES ELEMENTS D'INTERFACE (voir 15)
ri_gli=10
pen_lag=100

# PARAMETRES DE LA LOI ELASTOPLASTIQUE (valeurs tests)
young = 200000
poiss = 0.3
splas = 30
ecrou = 3577


PRE_GMSH()
MA0  =  LIRE_MAILLAGE()

# PASSAGE DU MAILLAGE EN QUADRATIQUE (voir 9 )
MA = CREA_MAILLAGE(MAILLAGE  = MA0, LINE_QUAD = _F(TOUT='OUI'))

# DEFINITION DES GROUPES
MA  =  DEFI_GROUP(
    reuse  = MA,
    MAILLAGE = MA,
    CREA_GROUP_MA = (

# LINES
     _F(NOM = 'DCB_G',  GROUP_MA = 'GM981'),     # LIEU DU CHARGEMENT IMPOSE
     _F(NOM = 'DCB_JB', GROUP_MA = 'GM984'),     # FACES POUR CONDITION DE SYMETRIE

# SURFACES
     _F(NOM = 'DCB1'  ,  GROUP_MA = 'GM988'),    # PARTIE VOLUMIQUE GAUCHE DE LA POUTRE
     _F(NOM = 'DCB2'  ,  GROUP_MA = 'GM989'),    # PARTIE VOLUMIQUE DROITE DE LA POUTRE
     _F(NOM = 'DCBJ'  ,  GROUP_MA = 'GM990'),    # ELEMENTS D'INTERFACE
   ),

    CREA_GROUP_NO = (
```


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

```
     _F(NOM = 'DCB_G',  GROUP_MA = 'GM981'),
     _F(NOM = 'NO_7',   GROUP_MA = 'GM954'),     # NOEUD BLOQUE DANS LA DIRECTION X
  )
)

# ORIENTATION DES ELEMENTS D'INTERFACE (VOIR 10 )
MA=MODI_MAILLAGE(reuse =MA,
                 MAILLAGE=MA,
                 ORIE_FISSURE =_ F(GROUP_MA= ' DCBJ '),
                 )

# AFFECTATION DES MODELISATIONS
MO=AFFE_MODELE(

                MAILLAGE=MA,
                AFFE= (
                _F(
                   GROUP_MA = ('DCB1','DCB2','DCB_G'),
                   PHENOMENE = 'MECANIQUE',
                   MODELISATION = 'D_PLAN'
                     ),
                _F(
                   GROUP_MA = 'DCBJ',
                   PHENOMENE = 'MECANIQUE',
                   MODELISATION = ' PLAN_INTERFACE '  # MODELISATION PLAN POUR LES ELEMENTS D'INTERFACE
                       ),
                 ))

# DEFINITION DE LA POSITION DE LA FISSURE DANS LE REPERE GLOBAL (voir 7)
ORIEN_FI=AFFE_CARA_ELEM(MODELE=MO,
                     MASSIF=(_F(GROUP_MA='DCBJ',
                            ANGL_REP=(90, 0 ,0)))); # EN 2D UNIQUEMENT LE PREMIER ANGLE

# DEFINITION ET AFFECTATION DU MATERIAU
# ICI UN SEUL MATERIAU, ON PEUT AUSSI CREER DEUX MATERIAUX DIFFERENTS L'UN POUR LE VOLUME ET L'AUTRE
# POUR LA FISSURE COHESIVE

MATE=DEFI_MATERIAU(
     ELAS=_F(
             E =  young,
             NU = poiss),
     ECRO_LINE=_F(
                  D_SIGM_EPSI=ecrou,
                  SY=splas),
     RUPT_FRAG = _F(                        # MOT CLE POUR LES CZM EN RUPTURE FRAGILE
                    GC        = tenac,      # PARAMETRE MATERIAU (voir 15)
                    SIGM_C    = sigc,       # PARAMETRE MATERIAU (voir 15)
                    PENA_LAGR = pen_lag,    # PARAMETRE NUMERIQUE (voir 15)
                    RIGI_GLIS = ri_gli,     # PARAMETRE NUMERIQUE(voir 15)
                   ))



CM=AFFE_MATERIAU(
                MAILLAGE=MA,
                AFFE=_F(
                        GROUP_MA = ('DCB1','DCB2','DCBJ'),
                        MATER = MATE),
               )

# CHARGEMENTS

SYMETRIE = AFFE_CHAR_MECA(
  MODELE = MO,
  FACE_IMPO = ( _F(GROUP_MA='DCB_JB',DY = 0)) # CONDITION DE SYMETRIE SUR LA LEVRE INFERIEURE
 )

TRACTION = AFFE_CHAR_MECA(
 MODELE = MO,
  DDL_IMPO = (_F(GROUP_NO = 'NO_7'   ,DX = 0)),  # BLOCAGE DU MODE RIGIDE
  FACE_IMPO= (_F(GROUP_MA = 'DCB_G'  ,DY = 1 ),  # DIRECTION DU CHARGEMENT
            ))

# DEFINITION DE LA LISTE D'INSTANT
```


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

```
deb=-(val_inf/delta_U)-0.0000001  # VALEUR DU DEBUT DES INSTANTS PERMETTANT D'ATTEINDRE UN NIVEAU
                                  # DE CHARGEMENT val_inf A L'INSTANT 0 PROGRESSIVEMENT
L_INST=DEFI_LIST_REEL(DEBUT   = deb,
     INTERVALLE=(
      _F( JUSQU_A = 0,        NOMBRE = 10 ),
      _F( JUSQU_A = tfin  ,   NOMBRE = nbpas ),

      ))
# ARCHIVAGE DE L'INSTANT 0 ET DES INSTANTS IMPAIRES (SOMMETS DE CHAQUE CYCLE)
INS_ARCH=DEFI_LIST_REEL(DEBUT   = deb,
     INTERVALLE=(
      _F( JUSQU_A = 0,        NOMBRE = 1 ),
      _F( JUSQU_A = 1,        NOMBRE = 1 ),
      _F( JUSQU_A = tfin  ,   NOMBRE = nbarch),

      ))

# DEFINITION DE L'EVOLUTION CYCLIQUE DU CHARGEMENT
#  a : FONCTION AMPLITUDE CONSTANTE AVEC SURCHARGE (voir 21)

a = []

for i in xrange(nb_cy_1) :
  a.extend((2*i,val_inf,2*i+1,val_sup))

for i in range(nb_cy_1,nb_cy_2) :
  a.extend((2*i,val_inf,2*i+1,val_surch))

for i in range(nb_cy_2,nb_cy_3) :
  a.extend((2*i,val_inf,2*i+1,val_sup))


FCT_FAT = DEFI_FONCTION(
  NOM_PARA = 'INST',
  PROL_GAUCHE='LINEAIRE',
  VALE     = a,
  )

DEFLIST =DEFI_LIST_INST(DEFI_LIST=_F(LIST_INST =L_INST ),
                 ECHEC=_F(
                 SUBD_METHODE='UNIFORME',
                 SUBD_PAS  = 3,
                 SUBD_PAS_MINI = 1.E-10,
                 SUBD_NIVEAU=10,
                                ),)

# CALCUL NON LINEAIRE

U=STAT_NON_LINE(
  MODELE     = MO,
  CHAM_MATER = CM,
  CARA_ELEM  = ORIEN_FI,   # AFFECTATION DES CARACTERISTIQUES D'ORIENTATION AUX ELEMENTS D'INTERFACE
  EXCIT      = (
                 _F(CHARGE = SYMETRIE),
                 _F(CHARGE = TRACTION, FONC_MULT = FCT_FAT),
               ),
  COMPORTEMENT  = (
                _F(RELATION = loi_vol,  GROUP_MA = ('DCB1','DCB2')),
                _F(RELATION = loi_czm , GROUP_MA = 'DCBJ'),
                ),
  INCREMENT  = _F(LIST_INST = DEFLIST,
                  INST_FIN = tfin,
                  ),
  NEWTON     = _F(MATRICE  = 'TANGENTE'  , REAC_ITER=1),
  CONVERGENCE = _F(
                   RESI_REFE_RELA = 1.E-6,
                   SIGM_REFE      = sigc,    # CONTRAINTE DE REFERENCE (voir 16)
                   DEPL_REFE      = dc,      # DEPLACEMENT DE REFERENCE (voir 16)
                   ITER_GLOB_MAXI = 8,
                  ),
  SOLVEUR = _F(
             METHODE = 'MUMPS', PCENT_PIVOT = 100 ), # SOLVEUR MUMPS NECESSAIRE POUR LES EI (voir 17 )
  ARCHIVAGE = _F(LIST_INST = INS_ARCH
    ))
```


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

```
# POST_TRAITEMENT : IMPRESSIONS DE L'INDICATEUR D'ENDOMMAGEMENT V3 ET DU POURCENTAGE D'ENERGIE DISSIPEE V4 (voir 22 )

IMPR_RESU(FORMAT='RESULTAT',RESU=(_F(RESULTAT=U,
                                     NOM_CHAM='VARI_ELGA',
                                     INST=tfin,
                                     NOM_CMP=('V3','V4'))))

FIN(FORMAT_HDF='OUI')
```


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

6 Conclusion

D'une manière générale, l'utilisation des éléments d'interface (EI) est recommandée quand plusieurs choix sont possibles. En effet, cette approche réunit les qualités des deux autres et répond le mieux aux exigences de robustesse, fiabilité et performance souhaitées pour des modèles destinés à terme à l'ingénierie. En dynamique, le modèle EI est en cours de validation, on lui préférera le modèle de joint. Ce dernier peut répondre à un certain nombre d'attentes pourvu qu'on l'utilise avec prudence. Enfin, l'élément à discontinuité interne a l'avantage de ne pas régulariser l'énergie, en revanche il cumule les inconvénients des autres approches. On conseille de ne l'utiliser que dans des cas de figure bien particulier, par exemple pour de la multi-fissuration dans une direction donnée *a priori* (voir [bib5]).

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

7 Bibliographie

[1] Debruyne G. et Laverne J. Propagation de fissure en dynamique, simulation numériques à l'aide d'éléments CZM, *CR-AMA-09.192*, 2009.

[2] Francfort G., Marigo J.-J. (1998) Revisiting brittle fracture as an energy minimization problem. *J. Mech. Phys. Solids* **46** (8), 1319-1342.

[3] Laverne J. Un modèle cohésif pour la fatigue, effet de surcharge et propagation 3D, *CR-AMA-09.038* , 2009.

[4] Laverne J., Rupture ductile avec les modèles cohésifs, *CR-AMA-09.074*, 2009.

[5] Laverne J. Formulation énergétique de la rupture par des modèles de force cohésives, *Thèse de Doctorat, Université Paris 13, Novembre 2004.*

[6] Lorentz E., Modélisation et simulation numérique de l'endommagement des structures. *Habilitation à Diriger les Recherches, Université Paris 6, Juillet 2008.*

[7] Rice J.R. The mechanics of earth quake rupture, *Proceding of international School of Physics pp. 555-649 (1990).*

[8] Lorentz E., ADELAHYD II - Recommandations relatives à la simulation numérique de zones cohésives par éléments finis d'interface, note interne EDF R&D 6125-1724-2021-01404 (2021).

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