

ABAQUS

Poutres

1 Rappel sur les unités

Dans Abaqus les unités ne sont pas entrées explicitement dans le code de calcul. C'est à l'utilisateur de respecter un système d'unités cohérent, les deux systèmes d'unités les plus courants étant donnés dans le Tableau 1.1.

Déplacement	Force	Pression	Masse
m	N	Pa	kg
mm	N	Mpa	t

TABLE 1.1 – Systèmes d'unités

2 Ressources en ligne

Beaucoup d'informations sur la simulation des poutres dans Abaqus sont disponibles dans :

- le [Getting Started](#) d'Abaqus.
- Le [Abaqus Analysis User's Guide](#). En particulier, on pourra faire des recherches sur :
 - [Truss elements](#)
 - [Beam elements](#)
 - [Spring elements](#)

3 Rappels

Forces résultantes et de moments On considère une surface Γ d'un solide \mathcal{S} soumise à une force surfacique de traction \vec{t} . La force résultante appliquée par le milieu extérieur sur \mathcal{S} est notée \vec{F} et le moment résultant exprimé en

un point A est noté $\vec{M}(A)$.

$$\vec{F} = \int_{\Gamma} \vec{t} \, d\Gamma,$$

$$\vec{M}(A) = \int_{\Gamma} \overrightarrow{AM} \wedge \vec{t} \, d\Gamma.$$

On note \vec{n} la normale extérieure à \mathcal{S} . L'équilibre de la structure au niveau de la surface Γ impose que $\vec{t} = \boldsymbol{\sigma} \cdot \vec{n}$. La force et le moment peuvent donc également se calculer en utilisant les formules suivantes :

$$\vec{F} = \int_{\Gamma} \boldsymbol{\sigma} \cdot \vec{n} \, d\Gamma,$$

$$\vec{M}(A) = \int_{\Gamma} \overrightarrow{AM} \wedge (\boldsymbol{\sigma} \cdot \vec{n}) \, d\Gamma.$$

Solides élancés Lorsqu'une dimension d'un solide est beaucoup plus grande que les autres, on dit que le solide est élancé. On peut alors appliquer la théorie des coques et la théorie des poutres, pour lesquelles le solide étudié est considéré de dimension 2 ou de dimension 1, respectivement. On considère une poutre dont la ligne moyenne est dirigée par l'axe \vec{x} . On isole l'ensemble des points de la poutre tels que $x < x_0$, $x_0 \in \mathcal{R}$. Le moment \vec{M} exercé sur cette partie de la poutre ($x < x_0$) par le reste de la poutre ($x > x_0$) s'écrit :

$$\vec{M}(A) = \mathcal{M}_t \vec{x} + \mathcal{M}_{fy} \vec{y} + \mathcal{M}_{fz} \vec{z},$$

\mathcal{M}_t étant le moment de torsion, et \mathcal{M}_{fy} et \mathcal{M}_{fz} étant les deux moments de flexion selon \vec{y} et \vec{z} , respectivement. Localement, la force surfacique $\vec{t} = \boldsymbol{\sigma} \cdot \vec{x}$ se décompose en :

$$\vec{t} = \sigma_{xx} \vec{x} + \vec{\tau}.$$

σ_{xx} représentant la contrainte de traction et $\vec{\tau}$ la contrainte de cisaillement.

TP1 : Poutres en traction

1 Ensemble de barres

1.1 Données du problème

Description On étudie un problème bidimensionnel composé de deux barres déformables notées **1** et **2**, de sections S_1 et S_2 , de modules d'Young E_1 et E_2 , et de masses négligeables. Les deux barres sont articulées en un point M de masse m situé à l'extrémité de la *Barre 2*, tel que représenté sur la Fig. 2.2. La *Barre 1* est articulée avec le *Bâti 0* au point O et la *Barre 2* est articulée avec le *Bâti 0* au point A . Le masse de M est noté m et l'accélération de la pesanteur est notée g .

Valeurs numériques :

- $E_1 = 1 \text{ Mpa,}$
- $E_2 = 3 \text{ Mpa,}$
- $S_1 = 1 \text{ cm}^2,$
- $S_2 = 2 \text{ cm}^2,$
- $m = 3 \text{ kg,}$
- $g = 9,8 \text{ m.s}^{-2}.$

Repérage On définit un repère $\mathcal{R}_0 = (O, \vec{x}_0, \vec{y}_0, \vec{z})$ associé à un référentiel supposé galiléen, \vec{y}_0 étant la direction verticale ascendante. L'origine O du repère se trouve à la surface du *sol 0* horizontal. On définit un repère $\mathcal{R}_1 = (O, \vec{x}_1, \vec{y}_1, \vec{z})$ de telle sorte que :

$$\vec{x}_1 = \frac{\overrightarrow{OM}}{\|\overrightarrow{OM}\|},$$

et un repère $\mathcal{R}_2 = (A, \vec{x}_2, \vec{y}_2, \vec{z})$ de telle sorte que :

$$\vec{x}_2 = \frac{\overrightarrow{AM}}{\|\overrightarrow{AM}\|}.$$

1.2 Étude géométrique

Paramétrage Les différentes longueurs initiales des barres (à chargement nul) sont définies comme suit :

$$\begin{aligned} L_0 &= \overrightarrow{OA} \cdot \vec{x}_0, \\ L_1 &= \|\overrightarrow{OM}\|, \\ L_2 &= \|\overrightarrow{AM}\|. \end{aligned}$$

Valeurs numériques :

- $L_0 = 5 \text{ m,}$
- $L_1 = 3 \text{ m,}$
- $L_2 = 4 \text{ m.}$

On note (x, y, z) les coordonnées cartésiennes dans \mathcal{R}_0 . Le point M se trouve dans la partie du plan telle que $y > 0$.

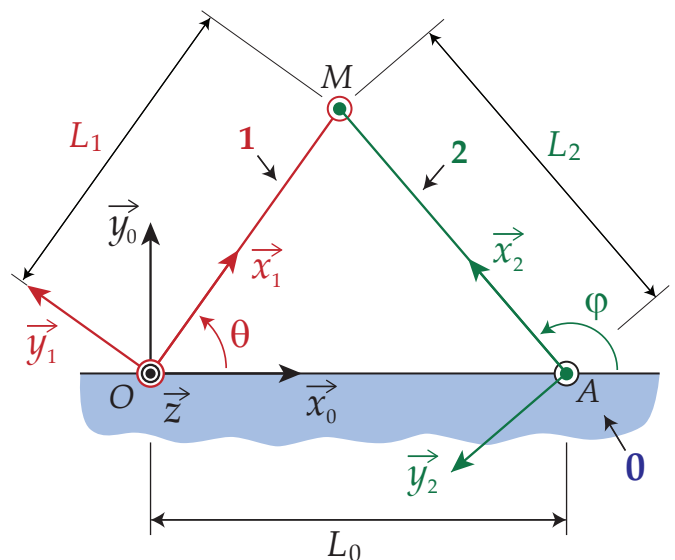


FIGURE 2.1 – Schéma cinématique de l'ensemble de barres

Graphe de structure Le graphe de structure est le suivant :

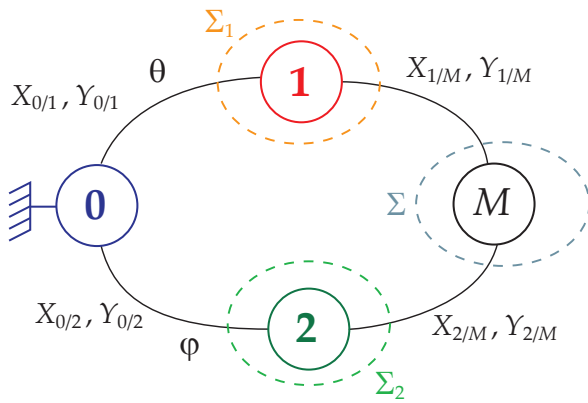


FIGURE 2.2 – Graphe de structure

Figure de changement de base La figure de changement de base est la suivante :

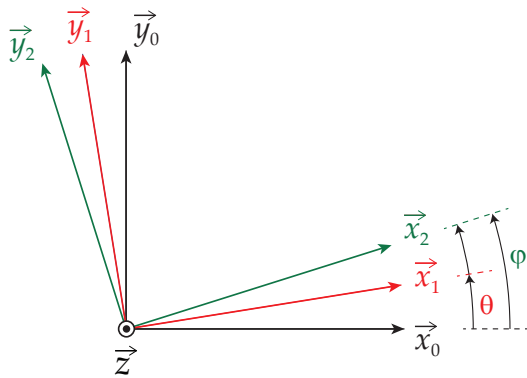


FIGURE 2.3 – Schéma cinématique de l'ensemble de barres

Résultat La fermeture de la boucle géométrique donne :

$$\cos(\varphi) = \frac{L_1^2 - L_2^2 - L_0^2}{2 L_0 L_2},$$

$$\cos(\theta) = \frac{L_0^2 + L_1^2 - L_2^2}{2 L_0 L_1}.$$

La vecteur position initial du point M est donc :

$$\overrightarrow{OM} = \begin{pmatrix} 1.8 \\ 2.4 \end{pmatrix} \quad (\text{en m}).$$

1.3 Étude statique (petites déformations)

Paramétrage Les forces exercées par $0/1$, $1/M$, $0/2$, et $2/M$ s'écrivent sous la forme :

$$\vec{F}_{0/1} = X_{0/1} \vec{x}_1 + Y_{0/1} \vec{y}_1,$$

$$\vec{F}_{1/M} = X_{1/M} \vec{x}_1 + Y_{1/M} \vec{y}_1,$$

$$\vec{F}_{0/2} = X_{0/2} \vec{x}_2 + Y_{0/2} \vec{y}_2,$$

$$\vec{F}_{2/M} = X_{2/M} \vec{x}_2 + Y_{2/M} \vec{y}_2.$$

Résultat L'isolement successif de chaque barre donne :

$$Y_{0/1} = Y_{1/M} = 0,$$

$$Y_{0/2} = Y_{2/M} = 0,$$

$$X_{0/1} = X_{1/M} = \frac{mg \cos \varphi}{\sin(\theta - \varphi)},$$

$$X_{0/2} = X_{2/M} = -\frac{mg \cos \theta}{\sin(\theta - \varphi)}.$$

On en déduit les longueurs l_1 et l_2 des barres après déformation notées :

$$l_1 = L_1 + \delta_1,$$

$$l_2 = L_2 + \delta_2.$$

Dans ces équations δ_1 et δ_2 représentent les allongements des barres vérifiant :

$$\delta_1 = -X_{0/1} / k_1,$$

$$\delta_2 = -X_{0/2} / k_2,$$

k_1 et k_2 représentant les raideurs des barres 1 et 2 de valeurs :

$$k_1 = E_1 S_1 / L_1,$$

$$k_2 = E_2 S_2 / L_2.$$

Les calculs numériques donnent :

$$\delta_1 = -0.7056,$$

$$\delta_2 = -0.1176.$$

Le vecteur déplacement \vec{u} du point M peut être approximé par :

$$\vec{u} = \delta_1 \vec{x}_1 + \delta_2 \vec{x}_2.$$

ce qui donne (en m) :

$$\vec{u} = \begin{pmatrix} -0.32928 \dots \\ -0.63504 \dots \end{pmatrix}.$$

1.4 Étude statique (grandes déformations)

Après déformation, les nouvelles longueurs des barres sont :

$$l_1 = 2.944 \text{ m},$$

$$l_2 = 4.1568 \text{ m}.$$

En recalculant la position du point M à partir des longueurs réactualisées des barres l_1 et l_2 , on obtient :

$$\vec{u} = \begin{pmatrix} -0.2808758 \dots \\ -0.6805427 \dots \end{pmatrix} \quad (\text{en m}).$$

Les angles θ et ϕ ayant changés, on peut recalculer les efforts $X_{0/1}$ et $X_{0/2}$, puis les longueurs des barres l_1 et l_2 , et enfin la position du point M . On obtient un nouveau vecteur déplacement \vec{u} s'écrivant :

$$\vec{u} = \begin{pmatrix} -0.3187138\dots \\ -0.7996676\dots \end{pmatrix} \quad (\text{en m}).$$

Ce processus itératif peut être répété autant de fois que l'on veut jusqu'à convergence vers la solution du problème. Après 200 itérations, le déplacement \vec{u} obtenu est :

$$\vec{u} = \begin{pmatrix} -0.3281184\dots \\ -0.8409010\dots \end{pmatrix} \quad (\text{en m}).$$

Lorsque les poutres sont comprimées, leurs sections augmentent et leurs longueurs diminuent. En prenant en compte de ce changement de géométrie dans les calculs de k_1 et k_2 (en faisant une hypothèse d'incompressibilité), on obtient après 200 itérations :

$$\vec{u} = \begin{pmatrix} -0.1950877\dots \\ -0.5266778\dots \end{pmatrix} \quad (\text{en m}).$$

2 Solution numérique

2.1 Méthode 1 : éléments *Truss*

Réaliser un calcul dans *Abaqus* pour simuler le problème de la déformation des deux barres en utilisant des éléments *Truss*. On comparera les résultats obtenus à ceux donnés par la théorie.

1. Dans le module *Part* :

- Créer un *Part* avec les options *Modeling Space* \rightarrow *2D Planar*, *Type* \rightarrow *Deformable*, *Base feature* \rightarrow *Wire*, et *Approximate size* \rightarrow 10.
- Un seul *Sketch* sera réalisé comportant la géométrie des deux barres **1** et **2**.
- Dans le *Sketch* commencer par créer un point fixe à l'origine du repère (O de coordonnées $x = 0$ et $y = 0$). Pour rendre le point fixe, utiliser l'outil *Add constraint* \rightarrow *Fixed*.
- En utilisant l'outil *Create Lines* : *Connected*, créer les segments $[OM]$ et $[MA]$. En utilisant l'outil *Add Dimension*, imposer la longueur de des segments $[OM]$, $[MA]$, et $[OA]$.

- En utilisant l'outil *Create Construction* : *Horizontal Line Thru Point*, créer la ligne de construction horizontale (OA). En utilisant l'outil *Add constraint* \rightarrow *Coincident*, imposer que le point A se trouve sur la ligne de construction horizontale.
- Pour vérifier que l'esquisse (*Sketch*) n'est ni sous-cotée ni sur-cotée, vérifier d'une part que les 2 barres et le trait de construction s'affichent bien en vert et d'autre part que l'outil *Drag entities* ne permet pas de modifier la position des objets.
- Finaliser l'esquisse (*Sketch*) en cliquant sur *Done* sous la fenêtre graphique.

2. Dans le module *Mesh*, créer un *Mesh*.

- Au dessus de la fenêtre graphique sélectionner l'option *Object* : *Part* afin que le maillage soit associé au *Part* et non à l'assemblage.
- Utiliser l'outil *Assign local seeds* avec les options *Method* \rightarrow *By number* et *Sizing Controls* \rightarrow *Number of elements* = 1 pour imposer la présence d'un seul élément-fini dans chaque barre.
- Utiliser l'outil *Assign Element Type* pour imposer les options *Family* \rightarrow *Truss*.
- Utiliser l'outil *Mesh Part* pour créer le maillage.

3. Dans le module *Property* :

- Créer un matériau *Material-1* pour la barre **1** et un autre *Material-2* pour la barre **2**.
 - On définira les propriétés de chaque matériau en créant un comportement de type *Mechanical* \rightarrow *Elasticity* \rightarrow *Elastic*.
 - Pour la *Young's Modulus*, on spécifiera la bonne valeur pour chaque barre. Si les distances de l'esquisse ont été spécifiées en mètre, alors l'unité des modules d'Young sera des Pascal Pa (dans le cas de longueurs en millimètres, spécifier des Mégapascals MPa).
 - Le coefficient de Poisson ν pourra être choisi nul (les éléments *Truss* ne se déforment qu'en traction-compression).
- Utiliser l'outil *Create Section* pour créer une *Section* pour chaque barre avec les options *Category* \rightarrow *Beam* et *Type* \rightarrow *Truss*. Pour chaque section, spécifier la bonne valeur du paramètre *Cross-sectional area* en fonction de la géométrie de la barre et du système d'unités choisi.
- Utiliser l'outil *Assign Section* pour créer des *Section Assignments* associant chaque section avec la bonne géométrie. Vérifier dans le *Part* que les *Section Assignments* ont bien été créés.

4. Dans le module *Assembly*, utiliser l'outil *Create Instance* pour créer un assemblage composé du *Part* comportant les 2 barres.

5. Dans le module *Step*, créer un *Step* de type *Static, General*. La durée totale du *Step* sera de 1 seconde (*Time period* = 1). Dans un premier le *Step* sera défini en petites déformations (*Nlgeom* = 1). On réalisera un calcul avec un seul pas de temps, ce qui peut s'obtenir dans l'onglet *Incrementation* en fixant le paramètre *Initial Increment size* à 1. Finalement, pour un calcul linéaire élastique, les paramètres par défaut du *Step* peuvent être conservés.
6. Dans le module *Load* :
- Utiliser l'outil *Create Boundary Condition* pour créer un déplacement imposé de type *Displacement/Rotation*.
 - Sélectionner à la souris les points *O* et *A*.
 - Bloquer les déplacements *U1* et *U2*. Notez que les éléments *Truss* ne comportant pas de degrés de liberté en rotation, cocher la case *UR3* n'aura aucun effet sur le calcul effectué.
 - Dans le module *Load*, utiliser l'outil *Create Load* pour appliquer la force de pesanteur appropriée au niveau du point *M*. On utilisera l'option *Types for Selected Step* → *Concentrated force*.
7. Dans le module *Job* :
- Utiliser l'outil *Create Job* pour créer un *Job*.
 - Faire clic-droit sur le *Job* dans l'arborescence de gauche et cliquer sur *Submit* pour lancer un calcul.
 - Si le calcul fonctionne bien, *Completed* s'affichera entre parenthèses à côté du *Job* dans l'arborescence de gauche une fois le calcul terminé. En cas d'erreur *Aborted* s'affichera entre parenthèse à côté du *Job*.
 - Si le calcul est *Aborted*, il faut comprendre pourquoi il n'a pas marché. Pour cela, on peut faire clic-droit sur le *Job* dans l'arborescence de gauche et choisir *Monitor*. On peut alors visualiser les erreurs (*errors*) et les avertissements (*warnings*). On peut noter les erreurs fréquentes ci-dessous :
 - Le *Section Assignment* n'a pas été fait.
 - Il s'affiche alors un message d'erreur du type : *X elements have missing property definitions. The elements have been identified in element set ErrElemMissingSection.*
 - Il est possible de voir les éléments pour lesquels le *Modèle* ne comporte pas de section. Pour cela, aller dans l'onglet *Results*, dans l'arborescence de gauche dérouler *Output Database* → *Job-1.odb* → *Element Sets* → *ErrElemMissingSection*. Les éléments pour lesquels la section est manquante s'afficheront alors en rouge.
 - Le type d'éléments-fini n'a pas été spécifié.
 - Le *Part* a été créé en sélectionnant l'option *3D* et non l'option *2D Planar*.
 - Les conditions aux limites en déplacement n'ont pas été imposées.
8. Dans le module *Visualization*, pour analyser les résultats, notez la valeur du déplacement du point *M* dans les directions \vec{x}_0 et \vec{y}_0 .
- Faire clic-droit sur le job et cliquer sur *Results*. Depuis l'onglet *Results*, utiliser l'outil *Create XY Data* et choisir l'option *ODB field output*.
 - Depuis l'onglet *Variables*, choisir *Position* → *Unique Nodal* et cocher *U1* et *U2* dans *U : Spatial displacement*.
 - Depuis l'onglet *Elements/Nodes*, sélectionner *Pick from viewport* et cliquer sur *Edit Selection* pour sélectionner le point *M*.
 - Cliquer sur *Save* ce qui aura pour effet de rajouter 2 lignes dans l'arborescence de gauche permettant aux déplacements suivant \vec{x}_0 et \vec{y}_0 du point *M*.
 - Vérifiez que vous obtenez bien $u_x = -0.32928$ m et $u_y = -0.63504$ m.
9. Essayer d'augmenter le nombre d'éléments dans l'une des deux *barres*. Que se passe-t-il ?
10. Relancer le calcul en se plaçant dans le cadre des hypothèses de grandes transformations.
- Aller dans le *Step*, onglet *Basic* et sélectionner l'option *Nlgeom* → *On*.
 - Relancer le calcul.
 - Faire clic-droit sur le *Job* et choisir *Monitor*. Notez le nombre total d'itérations avant convergence du calcul.
 - Aller dans l'onglet *Results*. Extraire les nouvelles valeurs des déplacements u_x et u_y du point *M*. Vérifiez que vous obtenez bien $u_x = -0.215138$ m et $u_y = -0.569759$ m.

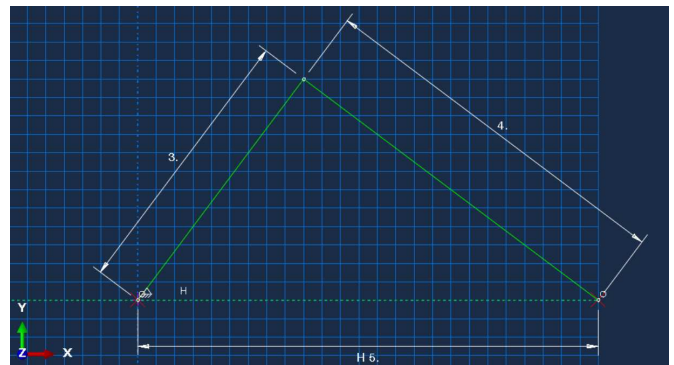


FIGURE 2.4 – Sketch

2.2 Méthode 2 : éléments *Beam*

On va maintenant réaliser un nouveau un calcul pour le même problème que précédemment en utilisant cette fois des éléments *Beam*.

- Créer un nouveau modèle *Model-2*. Pour cela, faire clic-droit sur *Models* dans l'arborescence de gauche et sélectionner *Create*.
- Dans le module *Part*, réaliser un *Part* pour chaque *barre*.

- (a) À la création des *Parts*, choisir les options *Modeling Space* → *2D Planar*, *Type* → *Deformable*, *Base feature* → *Wire*, et *Approximate size* → 10.
- (b) Faire attention non seulement à la longueur des barres **1** et **2** mais aussi à leur orientation et leur positionnement dans l'esquisse (*Sketch*), ce qui permettra de rendre leur assemblage plus facile. On pourra s'aider de points et de lignes de constructions afin d'obtenir le positionnement souhaité des barres.
3. Dans le module *Property* :
- (a) Créer un *Materials* pour chaque barre.
- (b) Créer un *Profile* de type *Rectangular* pour la barre **1** et la barre **2**. Choisir les longueurs des côtés de telle sorte que la section obtenue soit correcte.
- (c) Créer une *Section* pour chaque barre, en choisissant les options *Category* → *Beam* et *Type* → *Beam* (et non pas *Truss* comme dans le premier *Model*). Indiquer le *Profile* et le *Material* pour chaque barre.
- (d) Effectuer les *Section Assignments*.
- (e) Utiliser l'outil *Assign Beam Orientation* pour définir l'orientation de chaque barre en spécifiant la direction tangente à la poutre.
4. Dans le module *Mesh*, créer un *Mesh* associé au *Part* pour chaque barre.
- (a) Utiliser l'outil *Seed Edges* pour imposer la présence d'un seul élément dans chaque barre.
- (b) Utiliser l'outil *Assign Element Type* pour spécifier le type d'élément en choisissant les options *Geometric Order* → *Linear*, *Family* → *Beam* et *Element Controls* → *Shear-flexible* (poutre de Timoshenko).
5. Dans le module *Assembly* :
- (a) Créer une instance de chacune des 2 barres de type *Dependent (mesh on Part)*. Si les géométries des barres dans les *Parts* ont été bien définies par rapport au repère des esquisses, les deux barres se retrouvent automatiquement bien positionnées dans l'assemblage.
- (b) Utiliser l'outil *Translate Instance* afin de déplacer légèrement la barre **2** par rapport à la barre **1** pour qu'elles ne se touchent pas au niveau de la liaison *Pivot* en *M*. Pour cela on pourra utiliser l'outil *Translate instance* avec les options *start point X,Y* : 0.,0. et *end point X,Y* : 1.,0..
6. Dans le module *Step*, créer un *Step* en vue d'effectuer un calcul élastique.
7. Dans le module *Interaction* :
- (a) Utiliser l'outil *Create Wire Feature* pour créer un *Wire* entre les deux points extrémités des barres **1** et **2** devant se rencontrer au niveau de la liaison *Pivot* en *M*.
- (b) Créer un *Connector Sections* de type *Basic* → *Translation type* → *Cartesian*.
- Dans *Behavior Options*, créer dans le cadre *Behavior options* un objet de type *Elasticity*, *Definition* : *Rigid* et *Components* : *U1, U2, U3*.
- (c) Utiliser l'outil *Create connection assignement* pour créer un *Connection assignement* permettant d'associer le *Wire* et le *Connector Section*.
8. Retourner dans le module *Assembly* et placer les barres de telle sorte qu'elles soient jointives au point *M*.
9. Dans le module *Load*, imposer les conditions aux limites.
10. Dans le module *Job*, créer et lancer un *Job*.
11. Relancer le calcul en se plaçant dans le cadre de l'hypothèse des grandes transformations.
12. Essayer d'augmenter le nombre d'éléments dans chaque Barre. Que se passe-t-il ?
13. Créer un *Profile* de type *Generalized* pour la barre **2**. Pour éviter le message d'erreur à la création de la *Section*, on pourra cocher l'option *Section integration* → *Before analysis* pour éviter les erreurs à la création de la *Section*. On pourra donner une valeur quelconque aux paramètres *I11, I12, I22, J, Gamma O*, et *Gamma W*.

2.3 Méthode 3 : éléments Spring

L'objectif est maintenant de réaliser à nouveau un calcul en utilisant cette fois des éléments ressort (*Spring*). Pour définir des *Spring*, il n'est pas nécessaire de créer de *Part* ni d'utiliser le module *Property* (pas de *Material* ni de *Section* ou de *Section Assignment*). Ceci s'explique par le fait que les *Spring* sont définis directement à partir de leurs raideurs notées K_1 et K_2 , respectivement, pour les barres **1** et **2**.

$$K_1 = \frac{E_1 S_1}{L_1},$$

$$K_2 = \frac{E_2 S_2}{L_2}.$$

1. Dans le module *Assembly*, créer 3 *Reference Points*.
- (a) Aller dans le menu déroulant *Tools* → *Reference Point*.
- (b) Entrer successivement les coordonnées des 3 *Reference Points* à créer (0, 0, 0), (0, 5, 0) et (1.8, 2.4, 0) dans le cas d'unités en *m* pour créer les 3 *Reference Points*.
2. Dans le module *Interaction*, créer 2 *Springs*.
- (a) Dans l'arborescence de gauche faire clic-droit *Engineering Features* → *Springs/ Dashpots* et sélectionner *Create*.
- (b) Alternativement, on peut aussi créer le *Spring* depuis le menu déroulant supérieur *Special* → *Spring/Dashpots* → *Create* disponible uniquement depuis le module *Interaction*.
3. Dans le module *Step*, créer un *Step* en se plaçant dans le cadre de l'hypothèse des petites transformations.
4. Dans le module *Load*, imposer les conditions aux limites adéquates.

- (a) Aucun Part n'ayant été créé, la dimension de l'espace de travail n'a été spécifiée à aucun moment. Par défaut *Abaqus* considérera que l'espace de travail est *3D*. Pour se ramener à un problème plan, il faudra donc bloquer les déplacements dans la direction \vec{e}_z des 3 *Reference Points* en cochant la case *U3* dans les *BCs*.
- (b) Les autres conditions aux limites en déplacement et en effort sont les mêmes que pour les éléments *Truss* et les éléments *Beam*.
5. Dans le module *Job*, créer un *Job* et lancer un calcul.
 6. Relancer le calcul en se plaçant dans le cadre de l'hypothèse des grandes transformations.

TP2 : Treillis

1 Données du problème

Dans cet exercice, on utilisera des éléments *Truss*, afin de résoudre le problème de treillis représenté sur la Fig. 3.8.

Géométrie On supposera que toutes les barres composant le treillis sont reliées entre elles par des liaisons pivot et de section S avec :

$$S = 0.1 \text{ m}^2.$$

Matériau Le module d'Young E de chaque barre vérifie :

$$E = 210 \text{ GPa}.$$

Conditions aux limites Les barres du treillis sont en liaison pivot avec le bâti au point O et en liaison appui-plan de normale \vec{e}_y au point H .

Une force concentrée \vec{F}_3 est appliquée au point E avec :

$$\vec{F}_3 = -100 \vec{e}_y, \quad (\text{en kN}).$$

Une force linéique \vec{f} est appliquée le long des segments $[OA]$, $[AB]$, $[BC]$, $[CD]$, $[DE]$, $[EF]$, $[FG]$, et $[GH]$ avec :

$$\vec{f} = -1 \vec{e}_y, \quad (\text{en kN/m}).$$

Connaissant la section S des barres du treillis, on peut convertir cette force linéique en une force volumique \vec{f}_v en écrivant :

$$\vec{f}_v = \frac{\vec{f}}{S}.$$

2 Calcul dans Abaqus

Effectuer un calcul numérique de la déformation du Treillis dans *Abaqus*.

1. Dans le module *Part* :

(a) Créer un *Part* en cochant les options *2D Planar*, *Deformable*, *Wire*.

(b) Placer un point à l'origine du repère de l'esquisse et le fixer en utilisant une *Constraint*.

(c) Utiliser l'outil *Parameter Manager* pour créer une paramètre en spécifiant *Name = L1* et *Expression = 8*.

(d) Utiliser l'outil *Create Lines : Connected* pour créer la partie triangulaire gauche du treillis reliant les points O , A , I et J .

(e) Utiliser l'outil *Add dimension* pour ajouter les dimensions de cette partie du treillis jusqu'à ce que tous les traits apparaissent en vert. Éventuellement, utiliser l'outil *Add constraint* si cela est nécessaire.

(f) Utiliser l'outil *Edit Dimension Value* pour relier la valeur des dimensions en fonction du paramètre L_1 ayant été créée. Pour cela, cliquer sur une dimension de l'esquisse, dans la fenêtre qui s'affiche (*Edit Dimension*), cliquer sur $f(x)$, et dans la nouvelle fenêtre qui s'affiche *Parameter Manager* indiquer la valeur du paramètre (L_1 ou $L_1/2$ dans l'*Expression* de la dimension de la cote.

(g) Vérifier que le résultat obtenu ressemble à celui affiché sur la Fig. 3.1.

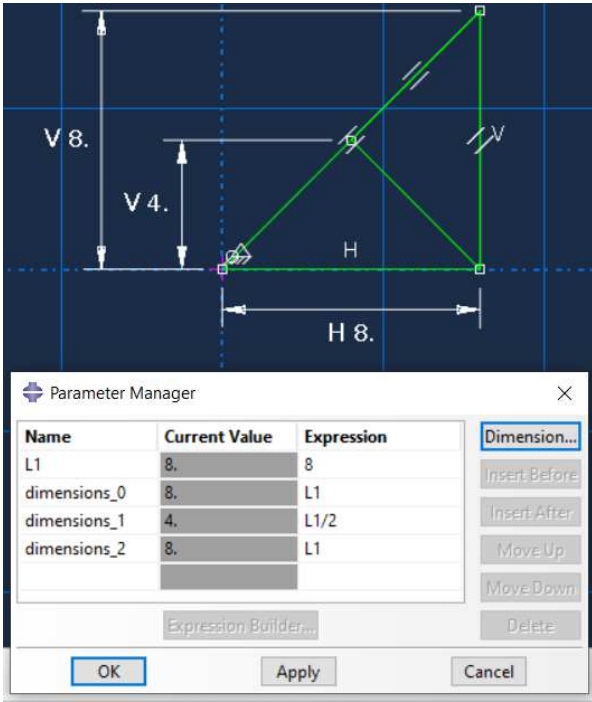


FIGURE 3.1 – Réalisation de l’esquisse pour l’extrémité triangulaire gauche du treillis

- (h) Utiliser l’outil *Create Lines : Connected* pour créer les barres [AB], [BC], [JB], [BL], [CL], [JK] et [KL].
- (i) Utiliser l’outil *Add dimension* pour ajouter les dimensions nécessaires.
- (j) Utiliser l’outil *Edit Dimension Value* pour exprimer les dimensions en fonction du paramètre L1.
- (k) Vérifier que le résultat obtenu ressemble à celui affiché sur la Fig. 3.2.

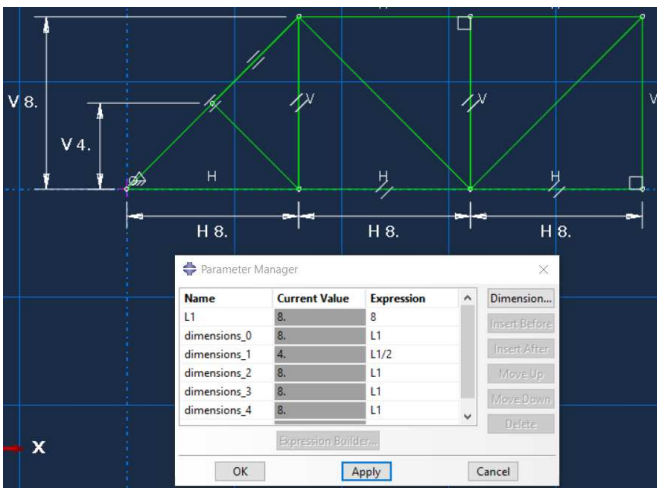


FIGURE 3.2 – Ajout de 2 cellules dans l’esquisse

- (l) Cliquer sur *Done* pour sauvegarder l’esquisse, puis éditer à nouveau l’esquisse en faisant clic-droit sur l’esquisse (*Squetch*) dans l’arborescence de gauche (*Part-1* → *Part-1* → *Features* → *Wire-1*) et en cliquant sur *Edit*.

- (m) Utiliser l’outil *Linear Pattern*, pour dupliquer 2 fois les segments [AB], [BC], [JB], [BL], [CL], [JK] et [KL] dans la direction \vec{e}_x . On pourra translater les segment dupliqué de 20 m vers la droite.
- (n) Vérifier que le résultat obtenu ressemble à celui affiché sur la Fig. 3.3.

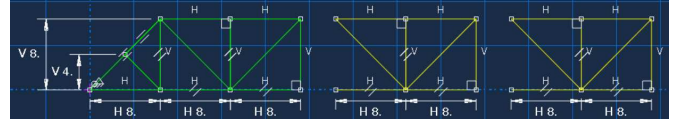


FIGURE 3.3 – Duplication des deux cellules dans l’esquisse (*Linear Pattern*)

- (o) Utiliser l’outil *Add Constraint* avec l’option *Coincident* pour raccorder les segments en C, E, N, et L.
- (p) Utiliser l’outil *Edit Dimension Value* pour exprimer les nouvelles dimensions en fonction du paramètre L1.
- (q) Vérifier que le résultat obtenu ressemble à celui affiché sur la Fig. 3.4.

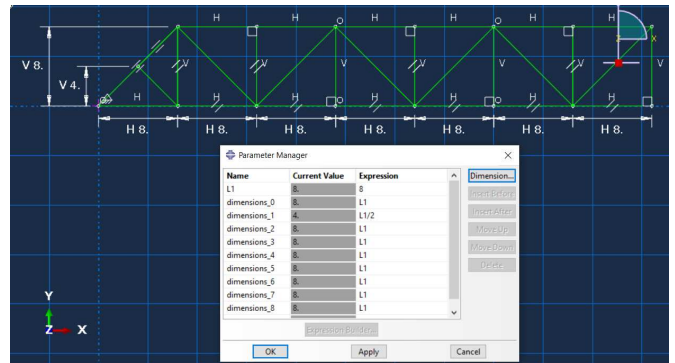


FIGURE 3.4 – Mise à jour des *Constraints* et des dimensions

- (r) Rajouter la partie triangulaire droite de treillis. Ne pas oublier d’imposer les *Constraints* et *Dimensions* pour que l’esquisse soit verte, et ne pas oublier non plus d’exprimer les dimensions en fonction du paramètre L1 dans le *Parameter Manager*.
- (s) Vérifier que le résultat obtenu ressemble à celui affiché sur la Fig. 3.5.

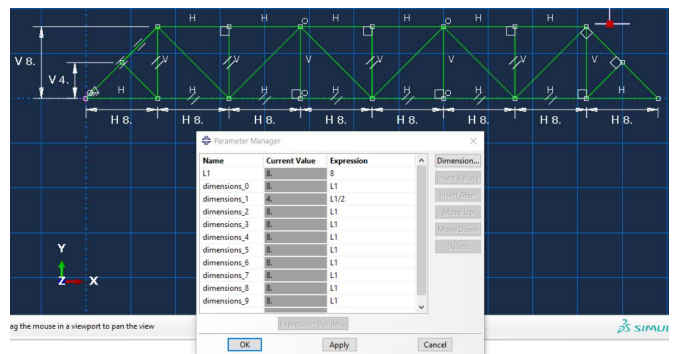


FIGURE 3.5 – Mise à jour des *Constraints* et des dimensions

- (t) Dans le *Parameter Manager* changer la valeur de $L1$ (mettre par exemple $L1 = 10$ m) pour vérifier que toutes les dimensions du treillis sont bien mises à jour.
 - (u) Cliquer sur *Done* pour sauvegarder l'esquisse.
2. Dans le module *Property*, créer le *Material*, la *Section* de type *Truss*, et effectuer le *Section Assignment*.
 3. Dans le module *Mesh*, créer le maillage dans le *Part* en imposant un élément par barre et en utilisant des éléments de type *Truss*.
 4. Dans le module *Step*, créer un *Step* en vue d'effectuer un calcul élastique en petite déformations.
 5. Dans le module *Load*, créer les conditions aux limites appropriées en déplacements et en efforts.
 - (a) La force \vec{F}_3 sera de type *Concentrated Force*.
 - (b) La force répartie \vec{f} sera de type *body force* car les forces de type *Line Load* ne sont pas compatibles avec les éléments *Truss*. Il faudra donc convertir la force linéique \vec{f} en une force volumique f_v .
 6. Dans le module *Job*, créer un *Job* et lancer le calcul.
 7. Dans le module *Visualization* :
 - (a) Utiliser l'outil *Plot contours on Deformed Shape* pour afficher la contrainte longitudinale $S11$ dans chaque barre sur la structure déformée. On pourra s'aider de la barre d'outils supérieure en sélectionnant dans les menus déroulants *Primary*, *S*, et *S11*.
 - (b) Utiliser l'outil *Common Plot Options* pour cocher dans l'encart *Deformation Scale Factor* l'option *Uniform* et *Value* : 5000.
 - (c) Utiliser l'outil *View* → *Graphic Options* du menu déroulant supérieur pour imposer un fond blanc avec l'option *Viewport Background*.
 - (d) Utiliser l'outil *View* → *ODB Display Options* du menu déroulant supérieur pour augmenter la section des barres à l'affichage.
Dans l'encart *Idealizations*, cocher la case *Render beam profiles* et spécifier *Scale factor* : 2.
 - (e) Vérifier que le résultat obtenu ressemble à celui affiché sur la Fig. 3.6.

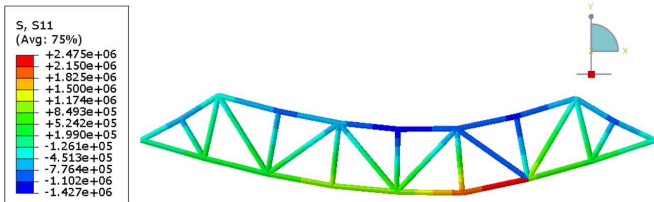


FIGURE 3.6 – Mise à jour des *Constraints* et des dimensions

- (f) Extraire les composantes du déplacement du point E sur lequel s'applique la force \vec{F}_3 .

$$u_3^x = -0.000275619 \text{ m},$$

$$= -0.275619 \text{ mm},$$

$$u_3^y = -0.00135068 \text{ m},$$

$$= -1.35068 \text{ mm},$$

- (g) Utiliser l'outil *Create XY Data* → *ODB field output* → *RF* : *Reaction force*. Extraire la valeur des composantes des efforts \vec{F}_1 et \vec{F}_2 .

$$\mathcal{F}_1^x = 0 \text{ N},$$

$$\mathcal{F}_1^y = 69500 \text{ N}.$$

$$\mathcal{F}_2^x = 0 \text{ N},$$

$$\mathcal{F}_2^y = 94500 \text{ N}.$$

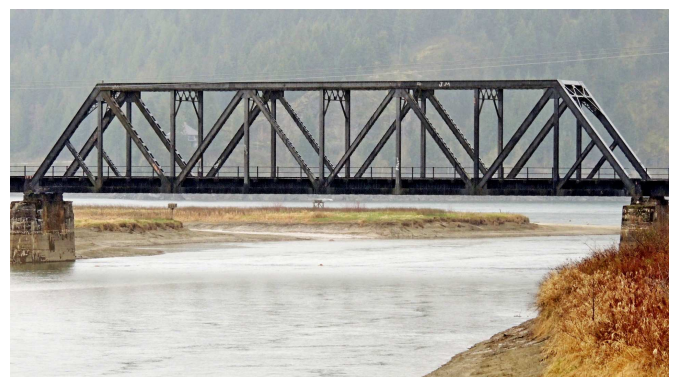


FIGURE 3.7 – Pont en treillis [source](#)

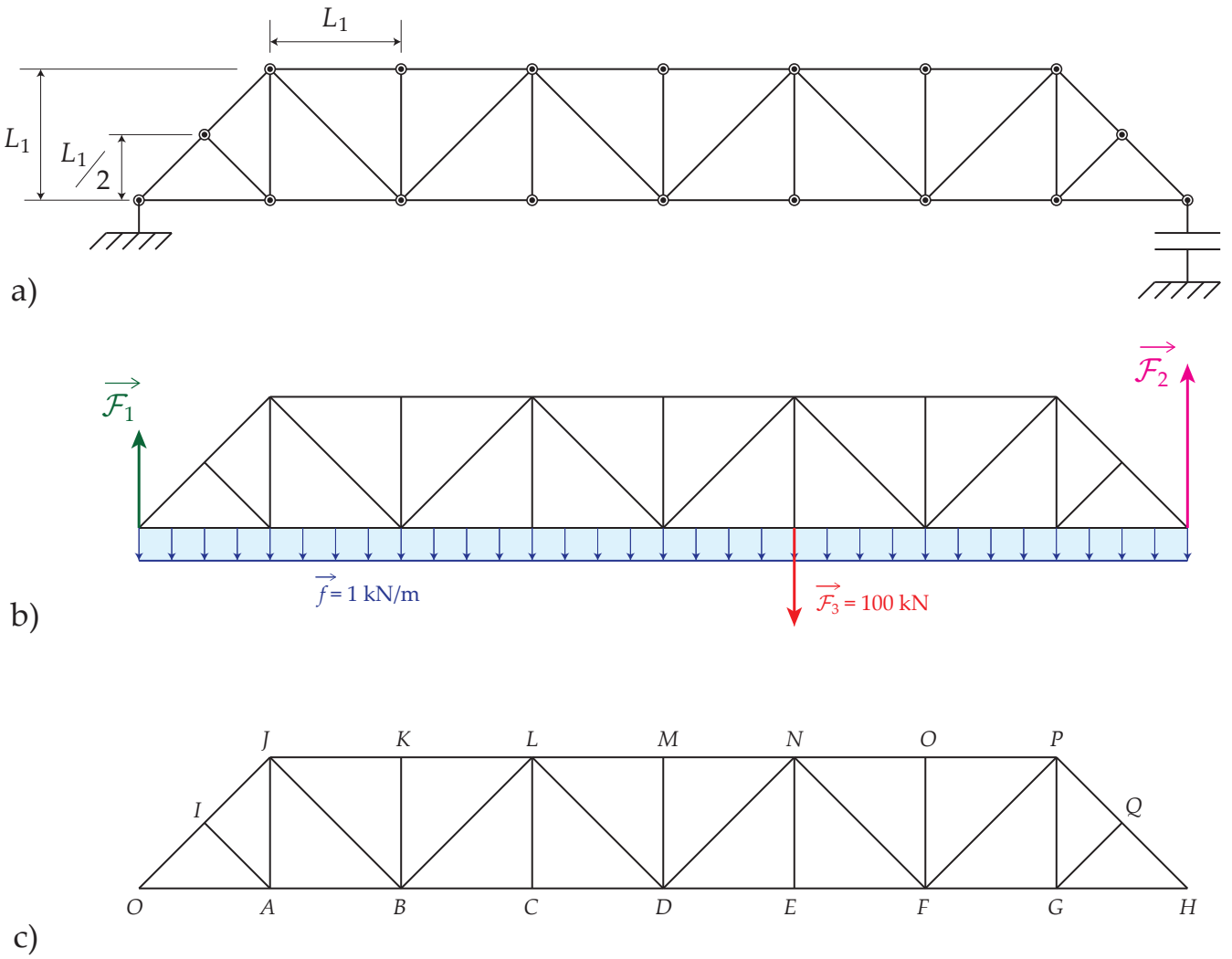


FIGURE 3.8 – Géométrie du treillis pour les calculs

TP3 : Poutres en flexion

1 Solution exacte

1.1 Cas 2D

Géométrie On s'intéresse au problème de flexion en contraintes planes représenté sur la Fig. 4.1a. Le point O est situé à l'origine du repère $\mathcal{R} = (O, \vec{e}_x, \vec{e}_y, \vec{e}_z)$. Les dimensions de la structures sont données par les paramètres L_x, L_y et L_z suivants :

$$\begin{aligned} L_x &= 100 \text{ mm}, \\ L_y &= 10 \text{ mm}, \\ L_z &= 10 \text{ mm}. \end{aligned}$$

Matériau Le matériau est de l'acier de module d'Young E et de coefficient de Poisson ν vérifiant :

$$\begin{aligned} E &= 210 \text{ GPa}, \\ \nu &= 0.3. \end{aligned}$$

Le moment de flexion de la poutre s'écrit :

$$\mathcal{I}_f = \frac{L_z L_y^3}{12}.$$

Conditions aux limites Sur le segment $[CF]$, les conditions au limites correspondent à une liaison *Appui-plan* et s'écrivent :

$$\begin{aligned} t_y &= 0, \\ u_x &= 0. \end{aligned}$$

u_x représentant la composante du déplacement suivant \vec{e}_x et t_y la composante de la traction suivant \vec{e}_y . On ajoute en plus la condition suivante au point O :

$$u_y = 0.$$

Les bords $[CD]$ et $[EF]$ sont libres d'effort. Le long du segment $[DE]$, les conditions aux limites s'écrivent :

$$\begin{aligned} t_x &= 0, \\ t_y &= -\frac{L_y^2/4 - y^2}{2 I_f} \mathcal{F}. \end{aligned} \tag{4.2}$$

Pour les calculs numériques on posera :

$$\mathcal{F} = 1 \text{ N}.$$

Pour simplifier la mise en donnée du calcul numérique, on peut supposer que l'effort est réparti de façon homogène le long du segment $[DE]$ et on écrit alors :

$$t_y = -\frac{\mathcal{F}}{L_y L_z}.$$

On peut également définir une force linéique homogène vérifiant :

$$t_y^{lin} = -\frac{\mathcal{F}}{L_y}.$$

Solution exacte La solution exacte du déplacement u_y le long de la ligne moyenne de la poutre (droite d'équation $y = 0$) est la suivante :

$$u_y = -\frac{(1 + 5\nu/4)(L_y)^2 x + (3L_x - x)x^2}{6 E \mathcal{I}_f} \mathcal{F}.$$

La solution théorique du champ de contraintes en un point de coordonnées (x, y) est la suivante :

$$\begin{aligned} \sigma_{xx} &= \frac{(L_x - x)y}{\mathcal{I}_f} \mathcal{F}, \\ \sigma_{yy} &= 0, \\ \sigma_{xy} &= -\frac{L_y^2/4 - y^2}{2 \mathcal{I}_f} \mathcal{F}. \end{aligned}$$

1.2 Cas 1D (Bernoulli)

Dans le cas d'un problème d'une poutre unidimensionnel (voir Fig. 4.1b), le déplacement vertical en bout de poutre s'écrit :

$$u_y = -\frac{(3L_x - x)x^2}{6 E \mathcal{I}_f} \mathcal{F}.$$

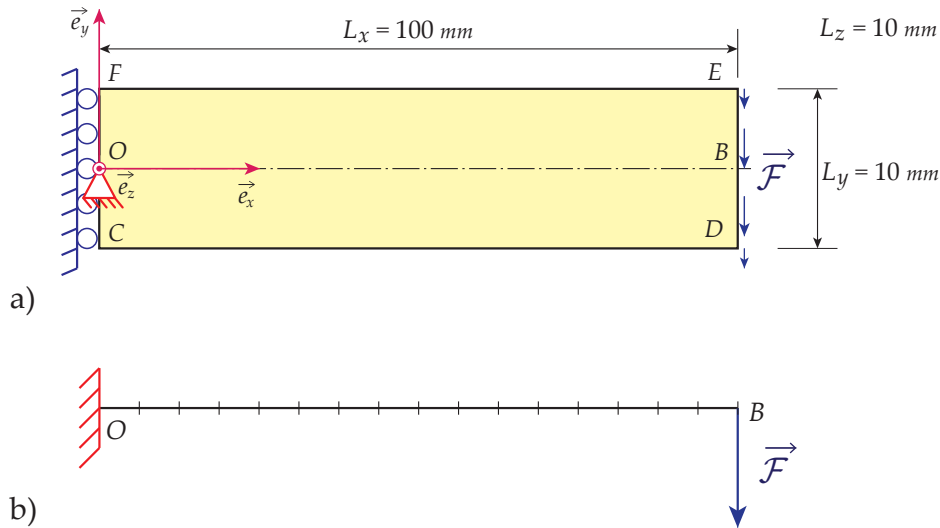


FIGURE 4.1 – Poutre en flexion dans le cas d’une modélisation a) bidimensionnelle ou b) unidimensionnelle

1.3 Tracé dans un tableur

1. Dans un tableur, tracer la déformée u_y en fonction de x obtenue en utilisant la solution exacte 2D et la solution 1D en utilisant la théorie de Bernoulli.

— Utiliser les premières lignes du tableau pour définir les paramètres numérique du problèmes : L_x , L_y , L_z , E , ν , F , et I_f .

— Dans une première colonne X du tableau, faire varier x de 0 à 100 avec un pas de 1.25 (permettant) de diviser le segment $[OB]$ en 80 segments de plus petite taille.

— Dans une deuxième colonne Uy_2D du tableau, entrer la formule permettant de calculer la solution exacte 2D du déplacement u_y le long de la fibre neutre de la poutre.

— Dans une troisième colonne Uy_B du tableau, entrer la formule permettant de calculer la solution approchée 1D du déplacement u_y le long de la fibre neutre de la poutre en utilisant la théorie de Bernoulli.

— Sur un même graphique, tracer les colonnes Uy_2D et Uy_B en fonction de X .

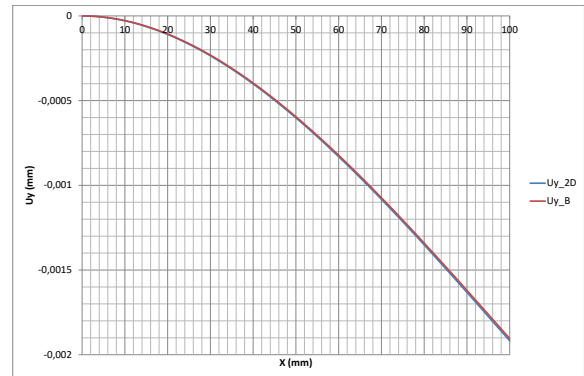


FIGURE 4.2 – Tracé de la flèche

— Dans la 4e colonne Err_B du tableau, calculer l’erreur $Err^{(B)}$ réalisée sur déplacement u_y calculé avec la théorie des poutres en prenant que la solution 2D comme solution de référence. On pourra se baser sur la formule suivante :

$$Err^{(B)} = \frac{|u_y^{(B)}(x) - u_y^{(2D)}(x)|}{|u_y^{(2D)}(L_x)|} \times 100,$$

$u_y^{(B)}$ désignant le champ de déplacement obtenu en utilisant la théorie de Bernoulli et $u_y^{(2D)}$ désignant le champ de déplacement exacte obtenu avec la théorie 2D contraintes planes.

— Sur un nouveau graphique, tracer la colonne Err_B en fonction de X .

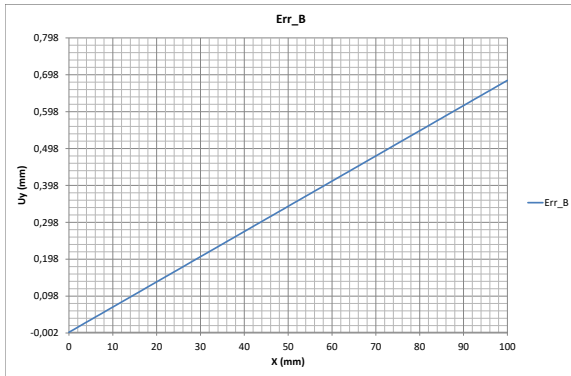


FIGURE 4.3 – Tracé de la flèche

2 Solution numérique

2.1 Cas 2D

Questions Résoudre le problème dans *Abaqus* pour $\mathcal{F} = 1$ N. On simplifiera les conditions aux limites en supposant une répartition homogène de la traction en bout de poutre. Le type d'élément utilisé sera des triangles. On effectuera un calcul en *Contraintes Planes*.

1. Dans le module *Part* :

- Créer un *Part* de type *2D Planar* et *Deformable*. Créer le point *O* au centre du segment $[CF]$ et fixer ce point à l'origine du repère.
- Réaliser le *Sketch* en plaçant toutes les *Constraints* et *Dimensions* de telle sorte que l'esquisse soit verte.
- Créer une partition séparant la partie supérieure de la poutre (telle que $y > 0$) de sa partie inférieure (telle que $y < 0$). Ceci permettra, lors de la création du maillage, qu'une série de nœuds soit placée le long du segment $[OB]$.
 - Utiliser le menu déroulant *Tools* → *Partitions* pour créer une *Partition*
 - Dans la fenêtre *Create Partition*, sélectionner *Type* → *Face* et *Method* → *Sketch*.
 - Dans le *Sketch*, créer le segment $[OB]$.

2. Dans le module *Property* :

- Utiliser l'outil *Create Material* pour créer le *Material* en indiquant les paramètres d'élasticité appropriés.
- Utiliser l'outil *Create Section* pour créer une *Section* avec les options *Category* → *Solid* et *Type* *Homogeneous*. On veillera à spécifier la bonne épaisseur (selon \vec{z}) dans la fenêtre *Edit Section*. Pour cela, cocher la case *Plane stress/strain thickness* et en indiquant la bonne valeur dans le système d'unité choisi.

(c) Créer le *Section Assignment*.

3. Dans le module *Mesh* :

- Choisir l'option *Object* : *Part* en vue de réaliser par la suite un assemblage de type *Dependent* : *Mesh on Part*.
- Utiliser l'outil *Seed Part* pour imposer une densité d'éléments de 1 mm.
- Utiliser l'outil *Seed Edges* pour imposer le long des bords du domaine rectangulaire :
 - 80 éléments le long des segments $[FE]$, $[OB]$, et $[CD]$.
 - 5 éléments le long des segments $[CO]$, $[OF]$, $[DB]$, et $[BE]$.
- Utiliser l'outil *Assign Mesh Controls* pour sélectionner les options de maillage *Element Shape* → *Tri*, *Technique* → *Free* et décocher case *Use mapped meshing where appropriate* dans l'encart *Algorithm*.
- Utiliser l'outil *Assign Element Type* pour imposer l'option *Family* → *Plane Stress*.
- Utiliser l'outil *Mesh Part* pour réaliser le maillage. Le résultat obtenu devrait ressembler à celui représenté sur la Fig. 4.4.

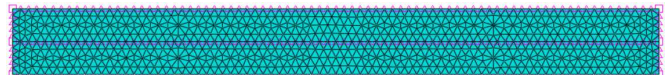


FIGURE 4.4 – Maillage

4. Dans le module *Step*, créer un *Step* en vue d'effectuer un calcul élastique en petites déformations.

5. Dans le module *Load* :

- Imposer le blocage suivant \vec{e}_x de tous les nœuds du segment $[CO]$.
- Imposer le blocage suivant \vec{e}_y du nœud *O*.
- Utiliser l'outil *Create Load* → *Surface traction* Imposer suivant le segment $[DE]$ une traction surfacique homogène de telle sorte que sa résultante soit égale à \mathcal{F} donné au début du sujet. Adapter la valeur de l'effort imposé au type choisi (linéique ou surfacique).

6. Dans le module *Vizualization*, extraire le déplacement vertical u_y de la ligne moyenne $[OB]$ dans un fichier texte :

- Créer un *Path* pour le segment $[OB]$:
 - Dans le menu déroulant *Tools*, sélectionner *Path* → *Create*.
 - Dans la fenêtre *Create Path*, choisir *Node list*.
 - Dans la fenêtre *Edit Node List Path*, sélectionner *Viewport selections* : *Add After...*
 - Sélectionner les nœuds correspondant dans le maillage aux points *O* et *B*.
- Extraire la déformée de la poutre :
 - Sélectionner l'outil *Create XYData* → *Path*.

- Sélectionner l'option *Model Shape* → *Undeformed*.
- Sélectionner l'option *Point Locations* → *Path points* et cocher la case *Include intersections*.
- Dans l'encart *X-Values*, sélectionner *X-distance*.
- Dans l'encart *X-Values*, cliquer sur *Field Output...*
 - Une fenêtre *Field Output* apparaît.
 - Dans l'encart *Output Variable*, sélectionner *U*.
 - Dans l'encart *Component*, sélectionner *U2*.
 - Exporter la déformée dans Excel en utilisant le menu *Plug-ins* → *Tools* → *Excel utilities*.
- Cliquer sur le bouton *Save As*.
- Dans l'arborescence de gauche, faire *clic-droit* → *Edit* sur le *XYData* qui s'est créé. Sélectionner les données du tableau qui s'affiche et les coller dans le tableau.
- Alternativement, on peut utiliser le menu déroulant *Plug-ins* → *Tools* → *Excel Utilities* si il est disponible.

- Choisir l'option *Magnitude = 1* (normalement le signe - a déjà été pris en compte lors de la définition de l'*Analytical Field*).

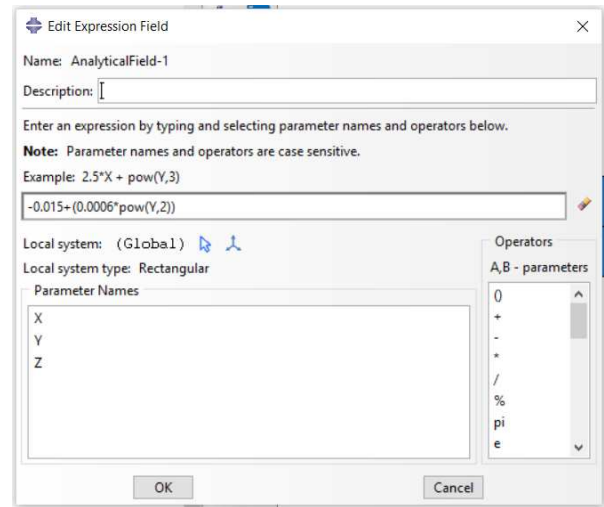


FIGURE 4.5 – Création du *Analytical field*

7. Dans le *tableur* :

- (a) Superposer la courbe numérique avec les courbes théoriques.
- (b) Tracer les erreurs en pourcentage des différentes déformées $u_y(x)$ par rapport au déplacement théorique 2D en bout de poutre.

8. Refaire le calcul en appliquant cette fois de façon exacte la charge en bout de poutre, conformément à l'Eq. (4.2).

- Dans l'arborescence de gauche, faire *clic-droit* sur le modèle et sélectionner *Copy-Model* pour copier le modèle.
- Renommer le nouveau modèle *Model-2D-exactBC*.
- Dans le module *Load* :
 - Supprimer le *Load* existant correspondant à un chargement homogène en bout de poutre.
 - Utiliser le menu déroulant *Tools* → *Analytical Field* → *Create* pour créer un *Analytical field*.
 - Dans l'encart *Type*, sélectionner *Expression Field* et cliquer sur *Continue*.
 - Entrer l'expression de la traction surfacique t_y telle que définie dans l'Eq. (4.2).
 - Utiliser l'outil *Create Load* pour créer un *Load* de type *Surface traction*.
 - Sélectionner les segments $[DB]$ et $[EB]$ puis cliquer sur le bouton *Done*.
 - Choisir l'option *Distribution* → *AnalyticalField-1*.
 - Dans l'encart direction choisir le vecteur $\vec{e}_y = (0, 1, 0)$

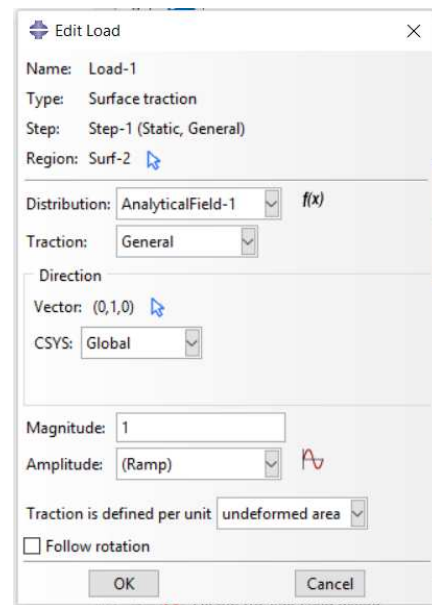


FIGURE 4.6 – Création du *Load*

- Compléter le fichier du *tableur* avec la nouvelle déformée de ligne moyenne obtenue. Compléter également le graphique donnant l'erreur en pourcentage par rapport à la solution analytique 2D.

2.2 Cas 1D - Bernoulli

Résoudre à nouveau le problème en utilisant cette fois des éléments *Beam*.

Questions

1. Dans le module *Part* :
 - (a) Créer un *Part* en vue de construire une géométrie 1D (la poutre en flexion) dans un espace en 2D.
 - (b) Dans le *Sketch*, dessiner la fibre neutre (= la ligne moyenne) de la poutre.
 - (c) Imposer les *Constraints* et les *Dimensions* pour que l'esquisse deviennent iso-contrainte. On rappelle que les traits sous-contraints sont jaunes, que les traits sur-contraints sont violets, et que les traits blancs sont les traits indiquant les dimensions.
2. Dans le module *Property* :
 - (a) Utiliser l'outil *Create Material* pour créer le *Material*.
 - (b) Utiliser l'outil *Create Profile* pour créer un *Profil* de type *Rectangular*.
 - (c) Utiliser l'outil *Create Section* pour créer une *Section* de avec les options *Category* → *Beam* et *Type* → *Beam*.
 - (d) Utiliser l'outil *Assign Section* pour relier la géométrie à la *Section*.
 - (e) Utiliser l'outil *Assign Beam Orientation* pour imposer l'orientation de la poutre.
 - (f) Pour plus d'informations sur l'orientation des poutres, consulter ce [lien](#).
3. Dans le module *Mesh* :
 - (a) Créer un maillage défini sur le *Part* (en vue de réaliser un assemblage de type *Dependent : Mesh on Part*).
 - (b) Utiliser l'outil *Seed Edges* pour imposer la génération de 80 éléments le long de la poutre.
 - (c) Utiliser l'outil *Assign Element Type* pour spécifier la présence d'éléments *Beam*.
 - (d) Dans l'encart *Element Controls* choisir l'option *Beam type* → *Cubic formulation*.
 - Quel est le *nom Abaqus* de l'élément utilisé ?
 - Consulter ce [lien](#).
 - Quelle est la signification de la lettre et des chiffres du *nom Abaqus* de l'élément ?
 - À quelle théorie de poutres correspond l'élément-finis utilisé ?
 - Consulter la liste des éléments *Beam* donnée sur ce [lien](#).
 - Quels sont les degrés de libertés actifs de l'élément utilisé ? On pourra se reporter à ce [lien](#) pour voir la signification de la numérotation des degrés de liberté.
4. Dans le module *Step*, créer un *Step* élastique en petite transformation.
5. Dans le module *Load* :
 - (a) Imposer les conditions aux limites en déplacement appropriées au niveau du point *O*.
 - (b) Imposer la condition aux limites en efforts appropriées au niveau du point *B*.
6. Dans le module *Job*, créer un *Job* et le lancer.
7. Dans le module *Visualization*, exporter le déplacement u_y obtenu le long de la poutre.
8. Dans le *tableur* :
 - (a) Ajouter une colonne U_B_Abq , avec la déformée obtenue.
 - (b) Calculer dans une nouvelle colonne l'erreur Err_B_Abq en prenant toujours comme référence la solution exacte 2D.
 - (c) Ajouter la courbe représentant cette erreur dans le graphique donnant les erreurs.

2.3 Cas 1D - Timoshenko

On va maintenant refaire le calcul en utilisant des éléments basés sur la théorie de Timoshenko :

1. Dupliquer le modèle.
2. Dans le nouveau modèle, changer le type de l'élément pour utiliser des éléments basés sur la théorie de Timoshenko.
3. Créer un *Job* et lancer le calcul.
4. Extraire les résultats du déplacement $u_y(x)$ de la poutre.
5. Compléter le fichier du tableur avec ces résultats.
6. Compléter la courbe des déformées et des erreurs avec les nouveaux résultats.

3 Éléments de correction

— Déplacement en bout de poutre théorique 2D :

$$-0,00191785714285714 \text{ mm}$$

— Déplacement en bout de poutre théorique 1D Bernouilli :

$$-0,0019047619047619 \text{ mm}$$

— Déplacement numérique 2D avec traction homogène :

$$-0,0018735 \text{ mm}$$

— Déplacement numérique 2D avec traction variable :

$$-0,00188291 \text{ mm}$$

— Déplacement numérique 1D, linear shear flexible :

$$-0,00191931 \text{ mm}$$

— Déplacement numérique 1D, cubic :

$$-0,0019047600 \text{ mm}$$

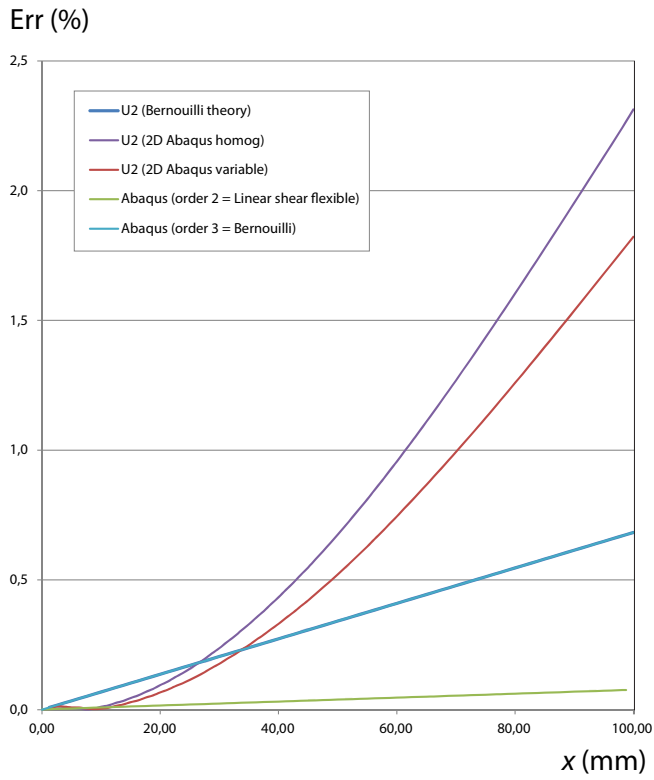


FIGURE 4.7 – Erreur en pourcentage de la flèche théorique en bout de poutre

TP4 : Poutres en torsion

1 Solution analytique

1.1 Solution 3D

On étudie le cylindre tridimensionnel creux et de répartition de masse homogène représenté sur la Fig. 5.5. La longueur du cylindre est notée L . Le rayon extérieur est noté R_{ext} et le rayon intérieur est noté R_{int} .

Paramètres géométriques Les paramètres géométriques sont les suivants :

$$\begin{aligned} L &= 0.2 \text{ m,} \\ R_{int} &= 0.1 \text{ m,} \\ R_{ext} &= 0.12 \text{ m.} \end{aligned}$$

Paramètres du matériau Les paramètres du matériau élastique sont :

$$\begin{aligned} E &= 3 \text{ GPa,} \\ \nu &= 0.35. \end{aligned}$$

Conditions aux limites Un encastrement est appliqué sur la surface d'équation $z = 0$. On a donc en coordonnées cylindriques :

$$\begin{aligned} u_r &= 0, \\ u_\theta &= 0, \\ u_z &= 0. \end{aligned}$$

On note \vec{t} la traction surfacique appliquée sur le reste du contour du domaine. La surface d'équation $\sqrt{y^2 + z^2} = R$ est un bord libre. On a donc en coordonnées cylindriques :

$$\begin{aligned} t_r &= 0, \\ t_\theta &= 0, \\ t_z &= 0. \end{aligned}$$

est appliquée sur la surface d'équation $z = L$ de telle sorte qu'un moment \mathcal{M}_t soit appliqué sur la surface d'équation $z = L$. On pose :

$$\mathcal{M}_t = 1 \text{ kN.m,}$$

et la traction surfacique \vec{t} s'obtient ensuite en coordonnées cylindriques en écrivant :

$$\begin{aligned} t_r &= 0, \\ t_\theta &= \frac{2\mathcal{M}_t}{\pi(R_{ext}^4 - R_{int}^4)} r, \\ t_z &= 0. \end{aligned}$$

On peut ensuite calculer le champ de déplacement solution du problème mécanique s'écrivant :

$$\begin{aligned} u_r &= 0, \\ u_\theta &= \alpha z r, \\ u_z &= 0, \end{aligned}$$

avec,

$$\alpha = \frac{\mathcal{M}_t}{G\mathcal{I}_t}.$$

avec \mathcal{I}_t le moment d'inertie de la poutre en torsion et G le module de Coulomb défini par :

$$\begin{aligned} \mathcal{I}_t &= \frac{\pi(D_{ext}^4 - D_{int}^4)}{32}, \\ G &= \frac{E}{2(1+\nu)}, \end{aligned}$$

avec D_{ext} et D_{int} les diamètres extérieurs et intérieurs de la poutre, respectivement.

Questions

- Dans un tableur, calculer pour la solution analytique :
 - L'angle rotation en torsion en bout de poutre.
 - De la norme du déplacement en bout de poutre en un point du rayon intérieur.
 - De la norme du déplacement en bout de poutre en un point du rayon extérieur.

1.2 Solution 1D

Dans le cas étudié, la solution 1D obtenue avec la théorie des poutres est identique à la solution tridimensionnelle.

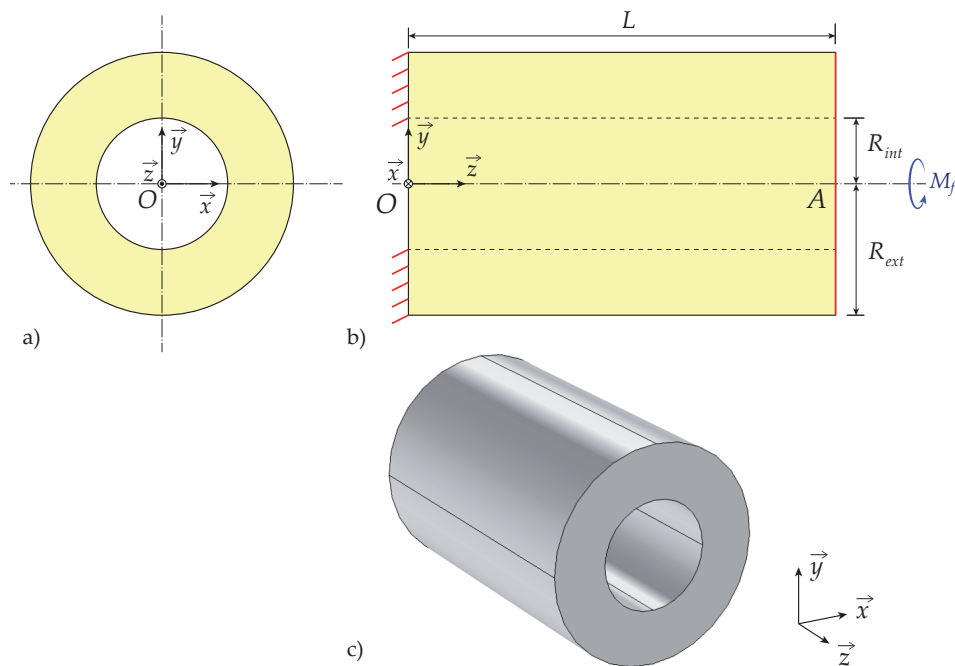


FIGURE 5.2 – Cylindre creux en torsion

2 Solution numérique

2.1 Calcul numérique 3D

Questions

- Dans le module *Part* :
 - Créer un *Part* de type *Solid* et *Deformable* dans un espace 3D en sélectionnant l'option *Type* → *Extrusion*.
 - Dans l'esquisse, dessiner la géométrie telle que vue dans le plan (O, \vec{x}, \vec{y}) .
 - Spécifier les *Constraints* et *Dimensions* jusqu'à ce que tous les traits de l'esquisse soient verts.
 - Terminer le *Part* en faisant l'extrusion.
- Dans le module *Mesh* :
 - Utiliser l'outil *Seed Part* pour spécifier une densité globale de 0.05 m.
 - Utiliser l'outil *Assign Mesh Controls* pour spécifier les options *Element Shape* → *Tet*, *Technique* → *Free*. Décocher l'option *Use mapped tri meshing on bounding faces where appropriate*.
- Dans le module *Property* :
 - Créer le *Material*.
 - Créer la *Section* de type *Homogeneous*.
 - Effectuer le *Section Assignment*.
- Dans le module *Assembly* :
 - Utiliser l'outil *Create Instance* pour créer une instance de Type *Dependent*.
- Dans le module *Step*, créer un *Step* en vue de réaliser un calcul élastique en petite transformation.
- Dans le module *Load* :
 - Utiliser le menu déroulant *Tools* → *Analytical field* → *Create* pour créer un *Analytical field* :
 - Dans l'encart *Type* de la fenêtre *Create Analytical Field*, choisir *Expression field* et cliquer sur *OK*.
 - Dans la fenêtre *Create Expression Field*, cliquer sur le bouton en forme pointeur à côté de *Local system*.
 - Cliquer sur le bouton *Datum CSYS List...*
 - Sélectionner le *Datum csys-1* ayant été créé précédemment.
 - De retour dans la fenêtre *Create Expression Field*, définir le champ de traction à appliquer en bout de poutre dans le repère cylindrique.
 - Le résultat devrait ressembler à celui indiqué sur la Fig. 5.3.
 - Utiliser l'outil *Create Boundary Condition* pour créer la condition aux limites d'encastrement sur le plan d'équation $z = 0$.
 - Utiliser l'outil *Create Load* pour imposer la traction \vec{t} en bout de poutre.
 - Dans l'encart *Types for Selected Step*, choisir l'option *Surface Traction*.

- Dans la fenêtre *Edit Load*, dans l'encart *Direction*, choisir l'option *CSYS* → *Picked* dans le menu déroulant.
- À droite du *CSYS*, cliquer sur le bouton avec le pointeur, puis cliquer sous la fenêtre graphique sur le bouton *Datum CSYS List ...* et sélectionner le système de coordonnées cylindrique *Datum csys-1* ayant été créé précédemment.
- Cliquer sur le bouton avec un pointeur se situant juste un peu au-dessus, à droite de *Vector before projection* : *Required*. Indiquer sous la fenêtre graphique les coordonnées des points extrémité (*starting point* et *end point*) du vecteur t_θ dans le repère cylindrique \mathcal{R}_θ .
- Choisir l'option *Distribution* → *AnalyticalField-1* pour pointer vers l'*Analytical Field* ayant été créé.
- Choisir l'option *Magnitude* = 1.

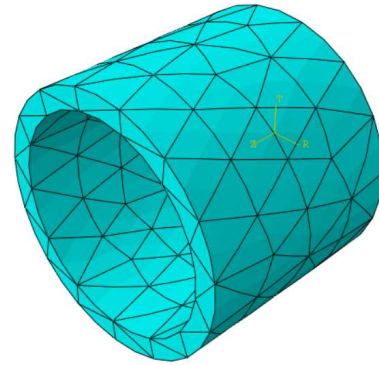


FIGURE 5.5 – Maillage obtenu

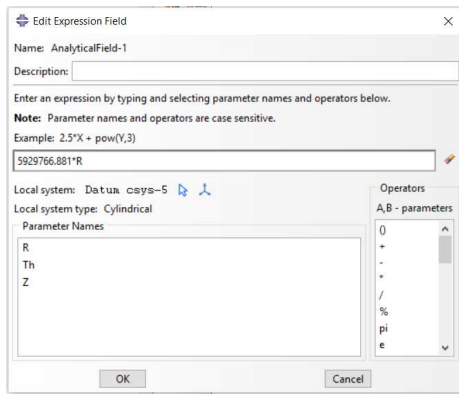


FIGURE 5.3 – Création du *Analytical field*

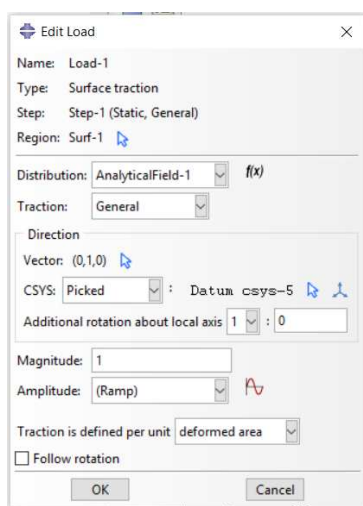


FIGURE 5.4 – Création du *Load*

2.2 Calcul numérique 1D

On cherche maintenant à comparer les résultats obtenus avec les calculs 3D avec ceux obtenus avec des éléments unidimensionnels.

Questions

1. Dans le module *Part* :
 - (a) Utiliser l'outil *Create Part* pour créer un *Part* en sélectionnant les options *Modeling Space* → *3D*, *Type* → *Deformable*, *Shape* → *Wire*, et *Approximate size* = 1.
 - (b) Créer la poutre dans l'esquisse, la coter et la dimensionner. Même si la poutre est orientée selon \vec{e}_z dans la réalité, on l'orientera selon \vec{e}_x dans le *Part*. On pourra réorienter la poutre dans le bon sens dans l'assemblage.
2. Dans le module *Property* :
 - (a) Utiliser l'outil *Create Material* pour définir le matériau.
 - (b) Utiliser l'outil *Create Profile* pour créer un *Profile*.
 - Choisir l'option de type *Pipe*.
 - Choisir l'option *Formulation* → *Thick-walled*.
 - Spécifier les paramètres géométriques (voir leur définition sur le dessin qui s'affiche).
 - (c) Utiliser l'outil *Create Section* pour créer la *Section*.
 - (d) Utiliser l'outil *Assign Section* pour associer la *Section* à la géométrie.
 - (e) Utiliser l'outil *Assign Beam Orientation* pour spécifier l'orientation de la poutre.
3. Dans le module *Mesh* :
 - (a) Créer un maillage dans le *Part* avec 100 éléments de *Timoschenko*.
4. Dans le module *Assembly* :
 - (a) Dans le module *Step*, créer un *Step* en vue d'effectuer un calcul élastique en petite déformation.
5. Dans le module *Load* :
 - (a) Imposer les conditions aux limites d'encastrement au point O .
 - (b) Appliquer le moment de torsion $\vec{\mathcal{M}}_t$ au point A .

7. Comparer le norme du déplacement en bout de poutre avec la solution théorique pour un point situé sur le rayon extérieur et un point situé sur le rayon intérieur.

6. Dans le module *Job*, créer un *Job* et lancer le calcul.
7. Dans le module *Visualization*, extraire la valeur de l'angle de rotation en torsion.
8. Comparer les valeurs de l'angle de rotation en bout de poutre et des déplacements intérieurs et extérieurs avec les valeurs théoriques. Calculer l'erreur en pourcentage par rapport à la solution théorique.
9. Dupliquer le *Model*. Refaire le calcul avec un *Profile* de type *Generalized*. Comparer les résultats avec les précédents.
10. Refaire le calcul en changeant le type d'éléments *Beam*, et comparer les résultats.
11. Refaire les calculs en changeant le nombre d'éléments le long de la poutre (en mettre 10), et comparer les résultats.

ABAQUS

Interaction - Contact

1 Problèmes avec contact

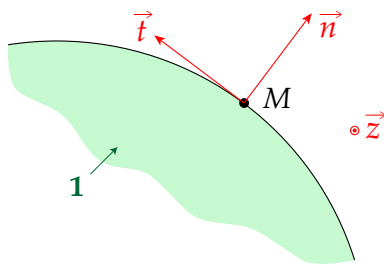


FIGURE 6.1 – Contact parfait entre un point et une pièce

Notations On considère un point M en contact avec un solide **1** comme représenté sur la Fig. 6.1. La normale extérieure à **1** au point M est notée \vec{n} . À partir de ce vecteur, on forme un repère orthonormal direct $\mathcal{R} = (M, \vec{n}, \vec{t}, \vec{z})$. La vitesse $\vec{v}_{M/1}$ et la force $\vec{F}_{1/M}$ sont de la forme (problème 2D) :

$$\begin{aligned} \vec{v}_{M/1} &= v_n \vec{n} + v_t \vec{t}, \\ \vec{F}_{1/M} &= N \vec{n} + T \vec{t}. \end{aligned}$$

Le non-décollement (maintien du contact entre M et **1**) impose :

$$\begin{aligned} v_n &= 0, \\ N &\geq 0. \end{aligned}$$

Contact parfait Le contact est parfait (= sans frottement) si :

$$T = 0.$$

Frottement de Coulomb (sec) On note α l'angle de glissement représenté sur la Fig. 6.2 et f le coefficient de frot-

tement, ces deux quantités étant reliées par l'équation :

$$f = \tan(\alpha).$$

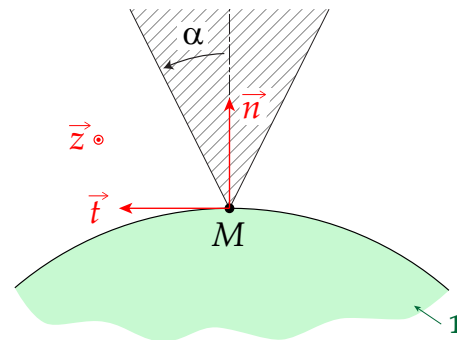


FIGURE 6.2 – Contact avec frottement entre un point et une pièce

On a en permanence :

$$\frac{|T|}{|N|} \leq f.$$

Cas du non-glissement :

$$\text{Si } \frac{|T|}{|N|} < f, \quad \text{alors } v_t = 0.$$

Cas du glissement :

$$\text{Si } v_t \neq 0, \quad \text{alors } \frac{|T|}{|N|} = f.$$

On peut écrire plus précisément :

$$\begin{aligned} \text{Si } v_t > 0, & \quad \text{alors } T = -fN. \\ \text{Si } v_t < 0, & \quad \text{alors } T = fN. \end{aligned}$$

Remarque : La force de frottement s'oppose au mouvement de $M/1$.

TP1 : Problèmes avec contact

Introduction

Pour rendre compte fidèlement des conditions aux limites réellement appliquées sur une pièce, il faut souvent modéliser les pièces avoisinant le système étudié et résoudre le problème de contact entre les différentes pièces. C'est ce type de problème que nous allons chercher à modéliser dans ce TP en partant d'un résultat d'essai mécanique. Les pièces de l'environnement seront définies comme des *corps rigides* de type *Analytical Rigid* ou *Discrete Rigid* et les contacts seront considérés soit parfaits soit avec frottement de Coulomb. Nous verrons ensuite comment extraire les résultats du calcul pour tracer les courbes caractérisant les efforts appliqués à la géométrie étudiée.

verticale d'un poinçon nommé *Punch* de forme cylindrique. Au début de l'essai, on fait descendre le *Punch* à une vitesse constante. Lorsque le *Punch* arrive en contact avec les bords de l'encoche, ceux-ci s'écartent progressivement. La pièce subit alors une contrainte maximale au niveau du fond de l'encoche en forme de demi-cercle. Lorsque cette contrainte dépasse une valeur caractéristique du matériau, une fissure se propage dans le matériau. Différentes variantes de cet essai peuvent être étudiées. Par exemple des trous peuvent être placés à différents endroits de l'éprouvette pour voir l'influence sur le trajet de fissuration, ou une pré-fissure peut être placée dans la continuité de l'encoche pour étudier l'influence sur la propagation stable ou instable de la fissure. Pour simplifier la modélisation du problème dans *Abaqus*, les 3 trous alignés ne seront pas pris en compte dans les simulations. De plus, on s'intéresse uniquement pour ce TP à la phase élastique de déformation précédant la rupture de l'éprouvette.

1 Problème étudié

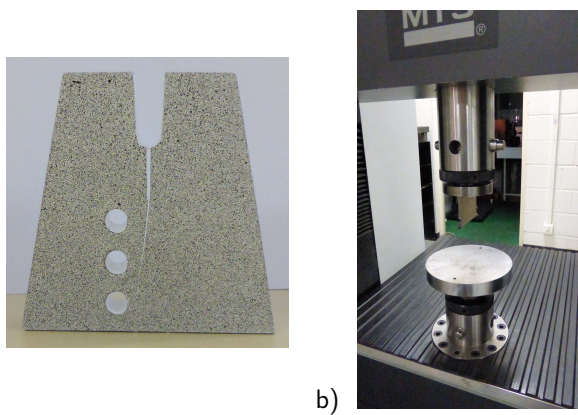


FIGURE 7.1 – Test 1 : a) Éprouvette trapézoïdale et b) Montage sur la machine de traction MTS de l'atelier de mécanique du LSPM.

Montage expérimental On étudie la rupture d'une éprouvette de forme trapézoïdale nommée *Trapezoid* similaire à celle représentée sur la Fig. 7.1a. Cette éprouvette permet d'étudier la rupture lorsqu'elle est placée dans le montage représenté sur la Fig. 7.1b. L'ensemble du montage est représenté schématiquement sur la Fig. 7.2. La pièce est posée sur un plateau de compression, l'encoche étant placée à la

Geométrie Le système étudié comporte 3 pièces nommées *Trapezoid*, *Punch* et *Ground*. La géométrie nominale de l'éprouvette trapézoïdale est représentée sur la Fig. 7.2. Cette éprouvette est découpée au laser dans une plaque de Plexiglas extrudée d'épaisseur 10 mm. Le rayon du faisceau laser étant d'approximativement 0.075 mm, le procédé de découpe induit un décalage de 0.075 mm des segments constituant le contour de la pièce étudiée. Pour prendre en compte ce décalage, la position des points de construction de l'éprouvette est modifiée comme indiqué dans le Tableau 7.1 dans la colonne de droite.

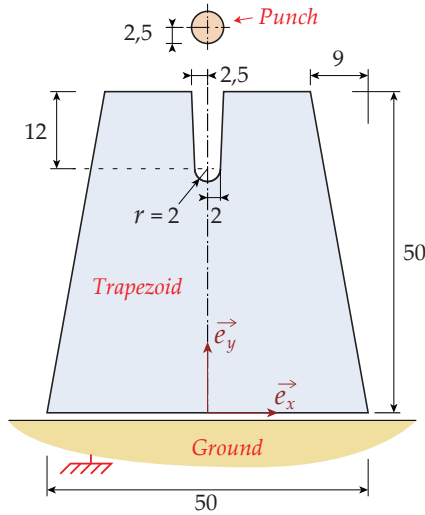


FIGURE 7.2 – Test 1 : Éprouvette en forme de trapèze (dimensions en mm)

va réaliser différents tests avec contact en faisant varier les options du calculs. Le Tableau 7.2 résume les différents tests et les paramètres spécifiques à chaque test.

Id	Type du <i>Punch</i>	Type de contact	f
A1	Analytical surface	Node-to-node	0
A2	Analytical surface	Node-to-node	0.45
A3	Analytical surface	Node-to-surface	0
A4	Analytical surface	Node-to-surface	0.45
B1	Discrete Rigid	Node-to-node	0
B2	Discrete Rigid	Node-to-node	0.45
B3	Discrete Rigid	Node-to-surface	0
B4	Discrete Rigid	Node-to-surface	0.45

TABLE 7.2 – Résumé des tests

Point	Coordonnées nominales (mm)	Coordonnées réelles (mm)
1	(0, 0)	(0.089705, 0.075)
2	(50, 0)	(49.9103, 0.075)
3	(41, 50)	(40.9373, 49.925)
4	(27.5, 50)	(27.5719, 49.925)
5	(27, 38)	(27.075, 37.9984)
6	(25, 36)	(25, 35.925)
7	(23, 38)	(22.925, 37.9984)
8	(22.5, 50)	(22.4281, 49.925)
9	(9, 50)	(9.0627, 49.925)

TABLE 7.1 – Coordonnées nominales et attendues des points

Matériau Le matériau est du PMMA (Polyméthacrylate de méthyle extrudé). Certains paramètres du matériaux sont donnés par le constructeur [4]. En particulier, les valeurs numériques du module d'Young E , du coefficient de Poisson ν et du frottement entre l'acier et le plastique f sont les suivantes :

$$\begin{aligned}
 E &= 3300 \text{ MPa,} \\
 \nu &= 0.37, \\
 f &= 0.45.
 \end{aligned}$$

Hypothèses pour les calculs On fera l'hypothèse des *contraintes planes* pour les calculs numériques. Le plateau de compression ainsi que le poinçon sont supposés rigides pour les calculs. Bien que le coefficient de frottement f entre l'acier et le PMMA soit de 0.45, on suppose dans un premier temps que le contact est parfait $f = 0$ avant d'effectuer les calculs pour $f = 0.45$. Pendant ce TP, on


2 Partie numérique

2.1 Cas d'un *Part Analytical Rigid*

Test A0

1. Renommer le modèle $A0$.
2. Dans le module du *Part* :

(a) Importer les *Parts* du *Trapezoid* et du *Ground* :

- Supprimer le *Model-1*.
- Copier le code JNL situé dans le fichier TP_trap.txt. Le coller dans la console située sous la fenêtre graphique d'*Abaqus*. Pour accéder à la console il faut cliquer sur l'icône  dans la fenêtre du bas.
- La géométrie du *Trapezoid* devrait s'afficher comme sur la Fig. 7.5.

(b) Création du *Punch* :

- Créer un *Part* pour le *Punch* en choisissant les options *Modeling Space* → *2D Planar*, *Type* → *Analytical rigid* et *Base Feature* → *Wire*.
- Dessiner un cercle dans une esquisse pour représenter le disque rigide. Pour un *Part* de type *Analytical rigid*, les arcs de cercle doivent être inférieurs à 180° , sinon un message d'erreur apparaît. Pour éviter ce problème, décomposer le cercle en 4 quarts de cercles. Pour spécifier plus facilement les cotes de la pièce, deux *Construction lines* horizontales peuvent être tracées, comme cela est représenté sur la Fig. 7.4. Idéalement, on placera le centre du cercle au point de coordonnées (25,50) pour faciliter la création de l'*Assembly*.

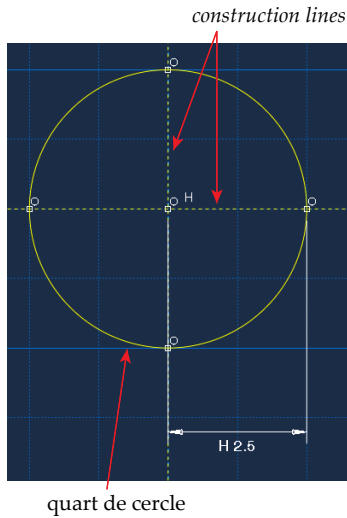


FIGURE 7.4 – Sketch permettant de définir le Punch

- Créer un *Reference Point* sur le cercle ou au centre du cercle.

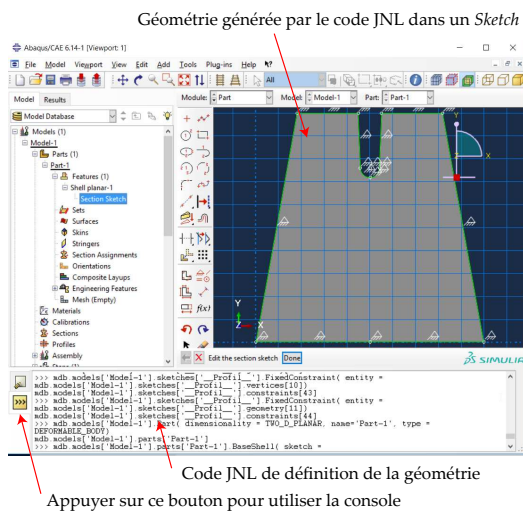


FIGURE 7.5 – Géométrie du Trapezoid créée avec le code JNL

3. Dans le module *Properties* :

- Créer un *Material* pour le Plexiglas composant l'éprouvette trapézoïdale.
- Créer une *Section* de type *Solid* → *Homogeneous* pour le *Trapezoid* en spécifiant la bonne épaisseur (10 mm).
- Créer un *Section Assignment* pour connecter la *Section* créée à la géométrie du *Trapezoid*.

4. Dans le module *Mesh*, créer un maillage composé d'éléments triangulaires avec une densité de 2 mm que l'on appellera maillage grossier (un maillage plus fin sera réalisé par la suite). Un maillage de ce type est représenté sur la Fig. 7.6.



FIGURE 7.6 – Maillage (grossier) du trapèze avec une densité de 2 mm

5. Dans le module *Assembly* :

- Créer une *Instance* du *Trapezoid* de type *Dependent*
- Créer une *Instance* du *Punch*.
- Si ce n'est pas déjà le cas, positionner le *Punch* en haut de l'encoche, le centre se trouvant au point de coordonnées (25,50) comme cela est représenté sur la Fig. 7.7.

Placer le centre du *Poinçon* au point de coordonnées (25, 50)

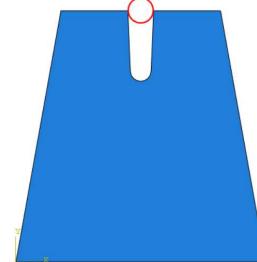


FIGURE 7.7 – Positionnement du *Punch* au niveau de l'entrée de l'encoche du *Trapezoid* dans le module *Assembly*

6. Dans le module *Step* :

- Créer un *Step* de type *Static, General* avec l'option *Incrementation* → *Fixed* et *Maximum number of increments* = 100 et *Increment size* = 0.01.
- Créer un *History Output Requests*, cocher *Contact* → *CFN, Total forces due to contact pressure and frictional stress*.
- Créer un *Field Output Requests*, cocher *Forces/Reactions RT, Reaction forces*.

7. Dans le module *Load* :

- Imposer une liaison appui-plan ($U_Y = 0$) sur le bord inférieur du *Trapezoid*.
- Imposer un déplacement de corps rigide sur le cercle rigide.
 - Créer une *Boundary Condition (BC)* de type *Displacement/Rotation* pour le *Reference Point* en choisissant $U_1 = 0$, $U_2 = -10$, et $U_3 = 0$.

8. Dans le module *Interaction*, imposer un contact parfait entre le cercle rigide et l'éprouvette.

- Créer une *Interaction Properties* de type *Contact* nommée *IntProp-Punch*. Choisir l'option *Mechanical* → *Normal Behavior* → "Hard" *Contact*.

- Créer une *Interaction Int-Punch* de type *Surface-to-Surface Contact* pointant vers cette *interaction propriétés* en choisissant :
 - le cercle comme *Master surface*,
 - les deux plans inclinés du *Trapezoid* comme *Slave surface* en cochant l'option *Surface to surface*.

9. Dans le module *Job* :

- (a) Créer un *Job* nommé *Job-A0*.
- (b) Lancer le *Job* puis faire clic-droit → *Results*.

10. Dans le module *Visualization* :

- (a) Choisir *Common Options* → *Uniform* → *Value* = 1 pour que le cercle s'affiche au niveau de la concentration de contrainte.
- (b) Tracer la courbe donnant la composante verticale de la force appliquée sur le *Punch* en fonction de la valeur absolue de son déplacement.
 - Dans *Results*, aller dans *XY Data* → *Create* puis choisir *ODB history output*. Dans l'onglet *Variables*, choisir *CFT2* pour extraire la force exercée par le *Trapezoid* sur l'éprouvette.
 - Aller à nouveau dans *XY Data* → *Create* puis choisir *Create XY Data*. Dans l'onglet *Variables*, choisir *Position* → *Unique Nodal* puis cocher *U2*. Dans l'onglet *Elements/Nodes*, choisir *Pick from viewport* et cliquer sur *Edit Selection* et cliquer sur le *Reference Point* du *Punch*.
 - Exporter les données dans un tableur. Pour extraire les valeurs des *XYData* dans un tableur, aller dans le menu *Plug-ins* → *Tools* → *Excel Utilities* et sélectionner les *XY data* voulant être exportés.

Test A1

1. Dupliquer le *Model A0* en faisant dans l'arborescence de gauche Clic-droit sur le nom du modèle puis *Copy Model*. Appeler *A1* le nouveau modèle.
2. Dans le module *Part* :
 - (a) Dans le *Part* du *Ground* :
 - Créer un *Reference Point* qui servira à appliquer les conditions aux limites d'encastrement sur le *Ground*.
 - Créer un *Set* pour ce *Reference Point*.
 - (b) Créer une partition du *Part Trapezoid* d'épaisseur 3 mm représentée sur la Fig. 7.8 afin de raffiner le maillage dans la zone de contact.

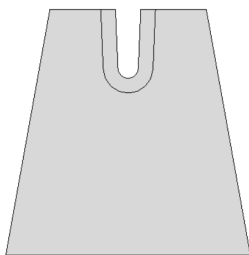


FIGURE 7.8 – Partitionnement du *Trapezoid*

On cherche à obtenir une densité de 1 mm sur la surface de contact.

- Dans le *Part*, depuis le menu déroulant supérieur, sélectionner *Tools* → *Partition*. Choisir *Type* → *Face* et *Method* → *Sketch*.
- Créer 2 *Construction Lines*, horizontales et verticales, se croisant au centre de l'arc de cercle du *Trapezoid*.
- Dans l'esquisse, créer une courbe contournant la zone de contact composée de 2 traits et de 2 arc de cercles comme cela a été fait sur la Fig. ??.

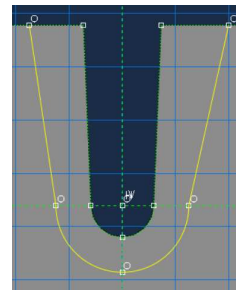


FIGURE 7.9 – Esquisse pour la partition du *Trapezoid*

- Imposer ensuite des *Constraints* en les traits de l'esquisse et les deux bords du *Trapezoid* en contact avec le *Punch*.
- Imposer un écart de 3 mm entre le rayon des arcs de cercles de l'esquisse et le rayon des arcs de cercles du *Trapezoid*. Le résultat devrait ressembler à celui représenté sur la Fig. ??.

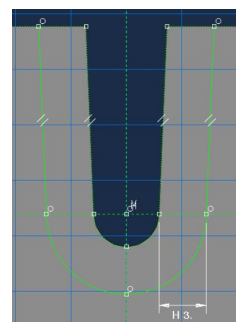


FIGURE 7.10 – Esquisse pour la partition du *Trapezoid*

3. Dans le module *Mesh* :

- (a) Définir une densité de 1 mm sur le contour de l'éprouvette se trouvant à l'intérieur de la *Partition* ayant été créée, comme cela est représenté sur la Fig. 7.11.

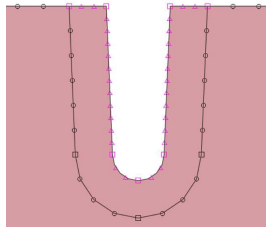


FIGURE 7.11 – Raffinement du *Trapezoid* au voisinage de la zone de contact

- (b) Créer un nouveau maillage pour l'éprouvette qui devrait ressembler à celui représenté sur la Fig. 7.12. On utilisera des éléments triangle non structurés sur tout le domaine.

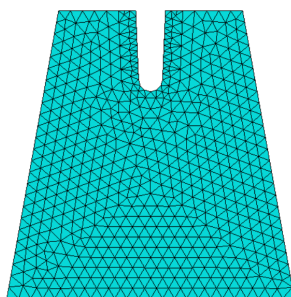


FIGURE 7.12 – Maillage du *Trapezoid*

4. Dans le module *Assembly* :
 - Ajouter le *Part Ground (Analytical Rigid)* à l'assemblage.
5. Dans le module *Interaction* :
 - Créer une *Interaction Property* nommée *IntProp-Ground* de type *Hard contact* entre le *Ground* et le *Trapezoid*.
 - Créer une *Interaction* nommée *Int-Ground* pour le contact entre le *Punch* et le *Trapezoid*. Le *Ground* étant de type *Analytical Surface* doit être défini comme étant la *Master Surface*.
6. Dans le module *Load* :
 - Supprimer la condition aux limites d'appui plan sur le bord inférieur.
 - Encastrent le *Ground* ($U_x = 0$, $U_y = 0$, et $R_z = 0$).
7. Dans le module *Job*, créer un *Job* nommé *Job-A1* pour le nouveau cas test.

Test A2

1. Reprendre le Test A1, en rajoutant dans *Interaction Properties* un *Tangential Behavior* avec une *Friction formulation* de type *Penalty* et le coefficient de frottement approprié.
2. Exporter la courbe force-déplacement dans le tableur. La superposer à la courbe des autres tests.

Test A3

1. Reprendre le Test A1, en utilisant une *Interaction* de type *Node-to-surface contact* (le coefficient de frottement f est ici égal à 0).
2. Exporter la courbe force-déplacement dans le tableur. La superposer à la courbe des autres tests.

Test A4

1. Reprendre le Test A3 avec un coefficient de frottement $f = 0.45$. Dans les options, on choisira une *Friction formulation* de type *Lagrange Multiplier*.
2. Exporter la courbe force-déplacement dans le tableur. La superposer à la courbe des autres tests.

Bilan Superposer les courbes Force-Déplacement obtenues dans *Excel*. Le graphique obtenu devrait ressembler à celui représenté sur la Fig. 7.7.

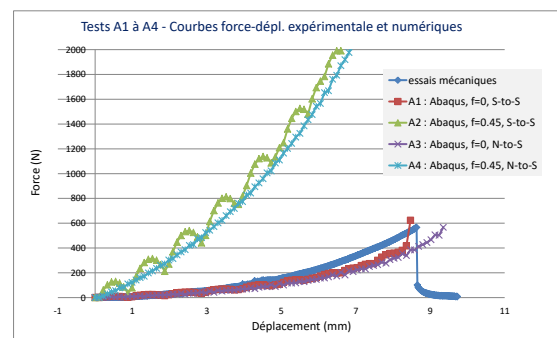


FIGURE 7.13 – Tests A1 à A4 : Courbes force-déplacement obtenue avec un corps rigide de type *Analytical* (S-to-S désigne *Surface-to-Surface* et N-to-S *Node-to-Surface*). Pour ces calculs la densité du maillage était de 1 mm.

2.2 Cas d'un *Part Discrete Rigid*

Dans cette Section, on reprend les Tests A1, A2, A3, et A4 de la Section 2.1 en utilisant pour le *Punch* un *Part* de type *Discrete Rigid*. Par rapport au *Part* de type *Analytical Rigid*, on peut faire les observations suivantes :

1. Un *Part* de type *Discrete Rigid* peut avoir des formes plus complexes qu'un *Part* de type *Analytical Rigid*. Par exemple, pour définir le *Punch*, il n'est pas nécessaire de la décomposer en quarts de cercles.

2. Le *Part* de type *Discrete Rigid* doit être maillé. On pourra choisir une densité de 0.2 mm pour tout les calculs.

3. Il faut définir un *Reference Point* dans le *Part* du *Punch* pour pouvoir lui appliquer les conditions aux limites.

4. Comme pour le *Part* de type *Analytical Rigid*, il faut bloquer une rotation lorsqu'on impose les conditions aux limites en déplacement.

5. Pour un *Part* de type *Discrete Rigid*, il n'y a pas besoin de définir de *Constraint* de type *Rigid Body*.

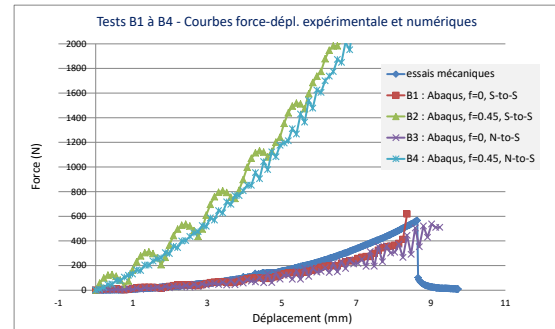


FIGURE 7.14 – Tests B1 à B4 : Courbes force-déplacement avec un corps rigide de type *Discrete* (voir la Fig. 7.13 pour la définition de S-to-S et N-to-S).

3 Conclusion

Le TP a permis de montrer comment imposer des conditions aux limites plus proches de la réalité dans *Abaqus* en utilisant des corps rigides et des *Interfaces* de type *Contact*. On a vu que la gestion du contact nécessite de respecter certaines règles de discrétisation. Notamment un contact de type *Node-to-Surface* requière un maillage fin au niveau des zones en contact.

Bibliographie

- [1] Amina DIEYE. Étude de la fissuration du plexiglas : Recherche de configurations stables et étude de l'influence des défauts. Stage de master 2. Université Paris 13. Soutenu le 14/09/2016.
- [2] PLEXIGLAS® GS/XT. Notice du fabricant EVO-NIK. <http://www.plexiglas.net/sites/lists/PM/DocumentsAP/211-1-PLEXIGLAS-GS-XT-en.pdf>

Annexe : Étude théorique

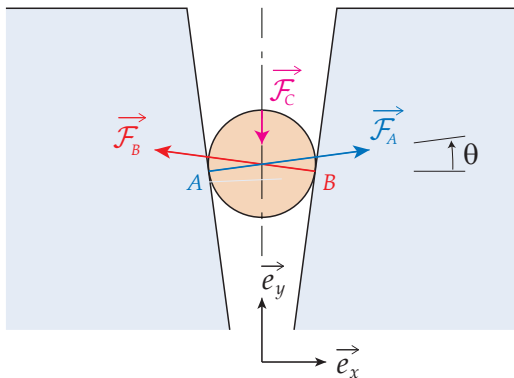


FIGURE 7.15 – Test 1 : Équilibre de l'appui rigide

Dans cette partie, on suppose que l'action du *Punch* sur le *Trapezoid* se traduit par des efforts ponctuels aux points A et B situés de part et d'autre de l'encoche à une hauteur $y = 44$ mm. L'isolement du disque rigide est représenté sur la Fig. 7.15. Le disque est soumis à une force \vec{F}_C imposée par la machine de traction/compression. L'éprouvette exerce sur le disque deux forces de réaction notées \vec{F}_A et \vec{F}_B . Les contacts *Punch/Trapezoid* étant supposés sans frottement, ces forces s'écrivent :

$$\begin{aligned}\vec{F}_A &= F_A (\cos \theta \vec{e}_x + \sin \theta \vec{e}_y), \\ \vec{F}_B &= F_B (-\cos \theta \vec{e}_x + \sin \theta \vec{e}_y), \\ \vec{F}_C &= -F_C \vec{e}_y.\end{aligned}$$

L'équilibre du disque en projection selon \vec{e}_x et \vec{e}_y s'écrit :

$$\begin{aligned}(\mathcal{F}_A - \mathcal{F}_B) \cos \theta &= 0, \\ \mathcal{F}_A \sin \theta + \mathcal{F}_B \sin \theta - \mathcal{F}_C &= 0,\end{aligned}$$

θ étant l'angle d'inclinaison des plans inclinés sur lesquels le disque est en appui (voir Fig. 7.15). Par conséquent,

$$\begin{aligned}\mathcal{F}_A &= \mathcal{F}_B, \\ \mathcal{F}_C &= 2 \mathcal{F}_A \sin(\theta).\end{aligned}$$

D'après la géométrie de la pièce, on a :

$$\begin{aligned}\tan \theta &= 1/24, \\ \theta &= \arctan(1/24).\end{aligned}$$

Les calculs donnent :

$$(\mathcal{F}_A)_x = 0.9991 \times \mathcal{F}_A, \quad (7.5a)$$

$$(\mathcal{F}_A)_y = 0.0416 \times \mathcal{F}_A, \quad (7.5b)$$

$$\mathcal{F}_C = 0.0833 \times \mathcal{F}_A. \quad (7.5c)$$

Les forces à appliquer au *Trapezoid* dans les simulations numériques sont $-\vec{F}_A$ en A et $-\vec{F}_B$ en B . L'Éq. (7.5c) permet de retrouver la force \mathcal{F}_C appliquée par la machine de traction/compression sur le disque.

Conditions aux limites avancées / homogénéisation
Travaux pratiques

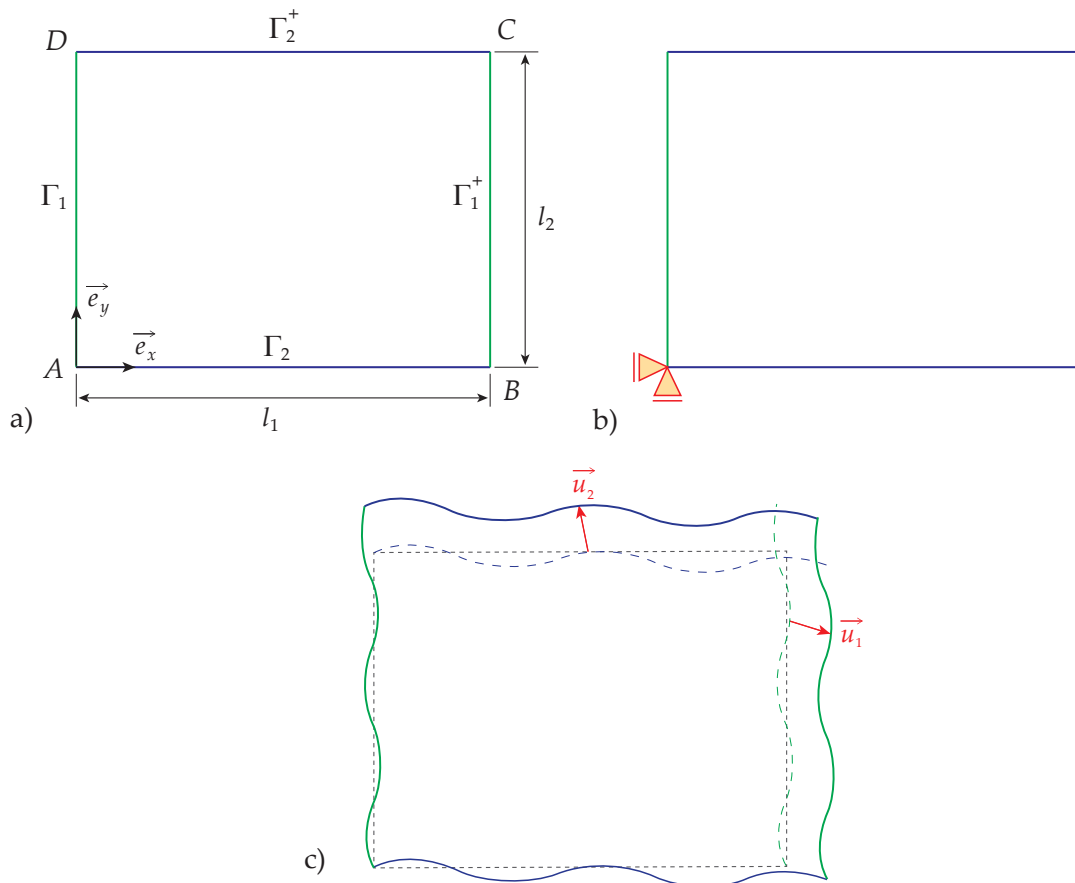


FIGURE 8.1 – a) Notations pour un motif rectangulaire et b) un champ de déplacement possiblement obtenu en imposant des conditions aux limites périodiques sur cette géométrie

Introduction

On considère une structure de *géométrie périodique*, ce qui signifie qu'elle est composée d'un *motif* se répétant pour former un *pavage*. Si on réalise un calcul numérique sur une structure composée d'un grand nombre de motifs, le calcul

risque de durer très le longtemps ou de dépasser la capacité de mémoire de l'ordinateur utilisé. On peut alors chercher à réaliser une simulation sur un seul motif en appliquant des conditions aux limites appropriées. Ce calcul peut ensuite servir à obtenir la matrice de Hooke \hat{C} du matériau homogénéisé :

$$\hat{\sigma} = \hat{C} \hat{\epsilon},$$

avec dans le cas bidimensionnel,

$$\hat{\sigma} = \frac{1}{S} \int_{\Omega} \sigma \, d\Omega, \quad (8.1)$$

$$\hat{\varepsilon} = \frac{1}{S} \int_{\Omega} \varepsilon \, d\Omega, \quad (8.2)$$

S représentant la surface totale du motif considéré (sans compter les vides). Le théorème de la divergence permet de transformer l'Éq. (8.2) en :

$$\hat{\varepsilon} = \frac{1}{S} \int_{\Gamma} (\mathbf{u} \otimes \mathbf{n})^s \, d\Gamma. \quad (8.3)$$

Pour la suite de la présentation, on considère un problème 2D composé d'une structure dont le motif est rectangulaire, comme le motif représenté sur la Fig. 8.1. D'autres géométries de motifs peuvent également être considérées comme des hexagones ou des parallélépipèdes pour les problèmes tridimensionnels.

1 Isolement d'un motif

Pour que les motifs déformés s'assemblent correctement les uns avec les autres dans le pavage, il faut que l'on ait :

$$\forall (x, y) \in \Gamma_1 : \quad \vec{u}(x+l_1, y) = \vec{u}(x, y) + \vec{u}_1, \quad (8.4a)$$

$$\forall (x, y) \in \Gamma_2 : \quad \vec{u}(x, y+l_2) = \vec{u}(x, y) + \vec{u}_2, \quad (8.4b)$$

avec Γ le contour du motif, l_1 la largeur du motif, l_2 sa hauteur, et \vec{u}_1 et \vec{u}_2 sont des vecteurs constants définis par : L'assemblage des motifs impose également les conditions suivantes sur le champ de contrainte σ :

$$\begin{aligned} \forall (x, y) \in \Gamma_1 : \quad & \sigma(x+l_1, y) \cdot \vec{u}_x = \sigma(x, y) \cdot \vec{u}_x, \\ \forall (x, y) \in \Gamma_2 : \quad & \sigma(x, y+l_2) \cdot \vec{e}_y = \sigma(x, y) \cdot \vec{e}_y. \end{aligned}$$

2 Conditions aux limites sur un motif

2.1 Objectif

L'objectif est de déterminer les conditions aux limites à appliquer sur le motif en imposant soit la valeur de la déformation moyenne $\hat{\varepsilon}$ soit la valeur de la contrainte moyenne $\hat{\sigma}$, les composantes de ces vecteurs étant notées :

$$\hat{\varepsilon} = \begin{pmatrix} \hat{\varepsilon}_{xx} \\ \hat{\varepsilon}_{yy} \\ 2\hat{\varepsilon}_{xy} \end{pmatrix}, \quad \text{et} \quad \hat{\sigma} = \begin{pmatrix} \hat{\sigma}_{xx} \\ \hat{\sigma}_{yy} \\ \hat{\sigma}_{xy} \end{pmatrix}.$$

2.2 CL de déplacement uniformes (KUBC)

Cette méthode s'appelle KUBC pour *Kinematically Uniform Boundary Conditions*. Avec cette méthode, on impose le champ de déplacement suivant sur tout le contour du motif :

$$\vec{u} = \hat{\varepsilon} \cdot \vec{x}.$$

2.3 CL de contraintes uniformes (SUBC)

Cette méthode s'appelle SUBC pour *Statically Uniform Boundary Conditions*. Les conditions aux limites appliquées s'écrivent :

$$\vec{t} = \hat{\sigma} \cdot \vec{n}.$$

\vec{n} étant la normale extérieure sortante. En détaillant pour chaque bord, on obtient :

$$\begin{aligned} (x, y) \in \Gamma_1 : \quad & \vec{t}(x, y) = -\hat{\sigma} \cdot \vec{e}_x, \\ (x, y) \in \Gamma_1^+ : \quad & \vec{t}(x, y) = \hat{\sigma} \cdot \vec{e}_x, \\ (x, y) \in \Gamma_2 : \quad & \vec{t}(x, y) = -\hat{\sigma} \cdot \vec{e}_y, \\ (x, y) \in \Gamma_2^+ : \quad & \vec{t}(x, y) = \hat{\sigma} \cdot \vec{e}_y. \end{aligned}$$

2.4 CL périodiques (PBC)

Cette méthode s'appelle PBC pour *Periodic Boundary Conditions*. Avec cette méthode, on applique directement les Eqs. (8.4a) et (8.4b) sur les nœuds en vis à vis de part et d'autre du motif étudié. En appliquant l'Eq. (8.3), on obtient :

$$\hat{\varepsilon} = \begin{pmatrix} u_{1x}/l_1 \\ u_{2y}/l_2 \\ u_{1y}/l_1 + u_{2x}/l_2 \end{pmatrix}.$$

On suppose que $u_{1y} = u_{2x}$ et on introduit les grandeurs \hat{u}_x , \hat{u}_y , et \hat{u}_{xy} définies par :

$$\begin{aligned} \hat{u}_x &= u_{1x}, \\ \hat{u}_y &= u_{2y}, \\ \hat{u}_{xy} &= u_{1y} = u_{2x}. \end{aligned}$$

On obtient alors :

$$\begin{aligned} \hat{u}_x &= \hat{\varepsilon}_{xx} l_1, \\ \hat{u}_y &= \hat{\varepsilon}_{yy} l_2, \\ \hat{u}_{xy} &= \frac{2 l_1 l_2}{l_1 + l_2} \hat{\varepsilon}_{xy}. \end{aligned}$$

Connaissant \hat{u}_x , \hat{u}_y , et \hat{u}_{xy} on peut facilement appliquer les conditions aux limites données dans les Eqs. (8.4a) et (8.4b).

2.5 Blocage des mouvements de corps rigides

Ces conditions aux limites ne suffisent pas à bloquer les mouvements de corps rigides du motif étudié. Il faut rajouter des appuis comme cela est fait sur le dessin de la Fig. 8.1.

3 Utilisation de la Toolbox *Homtools*

3.1 Principe de fonctionnement

la toolbox *Homtools* [1] permet d'appliquer des conditions aux limites sur des géométries rectangulaires dans *Abaqus*. Les degrés de liberté des nœuds sur le bord d'un motif sont éliminés et remplacés en 2D petites déformations par les 3 trois degrés de liberté suivants : $\hat{\epsilon}_{xx}$, $\hat{\epsilon}_{yy}$, et $\hat{\epsilon}_{xy}$. Ces degrés de liberté sont imposés par l'intermédiaire de deux *Reference Points* dont les coordonnées correspondent aux composantes de $\hat{\epsilon}$ en respectant la convention suivante :

RP1 :

$$U_1 \leftrightarrow \hat{\epsilon}_{xx},$$

$$U_2 \leftrightarrow \hat{\epsilon}_{yy}.$$

RP2 :

$$U_1 \leftrightarrow 2\hat{\epsilon}_{xy},$$

$$U_2 \leftrightarrow \text{non utilisé}.$$

Il reste à déterminer à quoi correspondent les forces appliquées sur les *Reference Points*. Conformément à la théorie de l'homogénéisation, le *Principe des Puissance Virtuelles* s'écrit pour un motif :

$$\mathcal{P}_{int}^* + \mathcal{P}_{ext}^* = 0,$$

avec,

$$\mathcal{P}_{int}^* = -S \hat{\sigma} : \hat{\epsilon}^*,$$

$$= -S (\hat{\sigma}_{xx} \hat{\epsilon}_{xx}^* + \hat{\sigma}_{yy} \hat{\epsilon}_{yy}^* + 2\hat{\sigma}_{xy} \hat{\epsilon}_{xy}^*),$$

et,

$$\mathcal{P}_{ext}^* = \mathcal{F}^T U^*,$$

$$\mathcal{P}_{ext}^* = \underbrace{\mathcal{F}_1 U_1^* + \mathcal{F}_2 U_2^*}_{RP1} + \underbrace{\mathcal{F}_1 U_1^*}_{RP2}.$$

Par identification, on obtient donc les forces nodales suivantes :

RP1 :	$\mathcal{F}_1 = S \sigma_{xx},$
RP2 :	$\mathcal{F}_2 = S \sigma_{yy},$
RP1 :	$\mathcal{F}_1 = S \sigma_{xy},$

Pour que le problème admette une solution unique, il reste à bloquer les mouvements de corps rigides, ce qui peut se faire en bloquant les déplacements selon \vec{e}_x et \vec{e}_y du point A :

$$u_x(A) = 0,$$

$$u_y(A) = 0.$$

3.2 Installation

1. Télécharger *Homtools* depuis la page <http://homtools.lma.cnrs-mrs.fr/>. voir aussi <http://homtools.lma.cnrs-mrs.fr/spip.php?article2#>
2. Décompresser *Homtools* dans un dossier de l'ordinateur, par exemple dans le *Dossier Personnel/homtools*.
3. Pour obtenir le chemin exact du dossier contenant la *toolbox*, ouvrir ce dossier, faire pointer la souris à l'intérieur de la fenêtre, faire *clic-droit* et choisir *Ouvrir dans un terminal* pour ouvrir un terminal. Taper la commande `pwd` et faire *Entree* pour voir s'afficher le chemin courant. Copier le chemin obtenu.
4. Dans *Dossier Personnel (Home)*, créer un fichier nommé *abaqus_v6.env* contenant uniquement une seule ligne de texte indiquant la direction du dossier contenant la *toolbox Homtools*. La ligne à écrire devrait ressembler à :


```
plugin_central_dir = "/export/home/users/...
master/master2/m2mec/XXXXXXXX/homtools0.9"
```

XXXXXXXX représentant la numéro étudiant du compte utilisé.
5. Ouvrir le *Dossier Personel*, faire pointer la souris à l'intérieur de la fenêtre, faire *clic-droit* et choisir *Ouvrir dans un terminal*. Depuis le terminal qui s'affiche taper la commande `Abaqus &` pour lancer *Abaqus* avec la *toolbox*.
6. Vérifier la bonne installation de la *toolbox Homtools* de la manière suivante :

- Aller dans le module *Part* et vérifier la présence du menu *Plug-ins* → *Homtools*, comme affiché sur la Fig. 8.2.
- Aller dans le module *Interaction* et vérifier la présence du menu *Plug-ins* → *Homtools*.
- Depuis les autres modules, il est normal que la *toolbox* n'apparaisse pas.

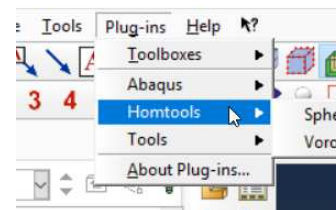


FIGURE 8.2 – Accès à *Homtools* depuis le menu *Plug-ins*

TP1 : Utilisation de *Reference Points*

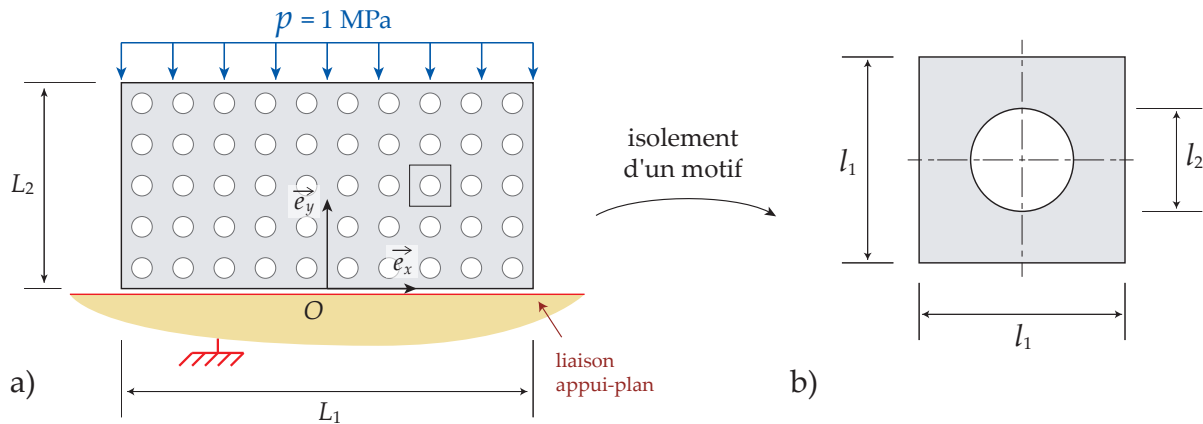


FIGURE 9.1 – a) Géométrie et conditions aux limites du problème étudié et b) zoom sur un motif du pavage de la structure

1 Dalle avec motif symétrique

1.1 Données du problème

Géométrie On étudie en 2D la déformation de la structure rectangulaire représentée sur la Fig. 9.1a. La largeur du rectangle est notée L_1 , sa hauteur L_2 , et son épaisseur e , avec :

$$\begin{aligned} L_1 &= 100 \text{ mm,} \\ L_2 &= 50 \text{ mm,} \\ e &= 1 \text{ mm.} \end{aligned}$$

La structure est percée de 50 (10×5) trous circulaires de diamètre d_1 répartis régulièrement sur la structure distants de l_1 dans la direction \vec{e}_x et l_2 dans la direction \vec{e}_y avec :

$$\begin{aligned} l_1 &= 10 \text{ mm,} \\ l_2 &= 5 \text{ mm.} \end{aligned}$$

La structure peut être reconstituée en pavant le motif représenté sur la Fig. 9.1b.

Matériau Le module d'Young E et le coefficient de Poisson ν du matériau prennent les valeurs suivantes :

$$\begin{aligned} E &= 210000 \text{ MPa,} \\ \nu &= 0.3. \end{aligned}$$

Conditions aux limites Le problème est supposé bidimensionnel et on se place dans le cadre de l'hypothèse des contraintes planes. La structure est soumise à un chargement homogène de compression sur sa face supérieure d'intensité p avec :

$$p = 1 \text{ MPa.}$$

La face inférieure est en contact sans frottement avec le Sol supposé rigide et horizontal. Les forces de pesanteur sont négligées pour toute l'étude.

1.2 Calcul global 1 (SY-GL1)

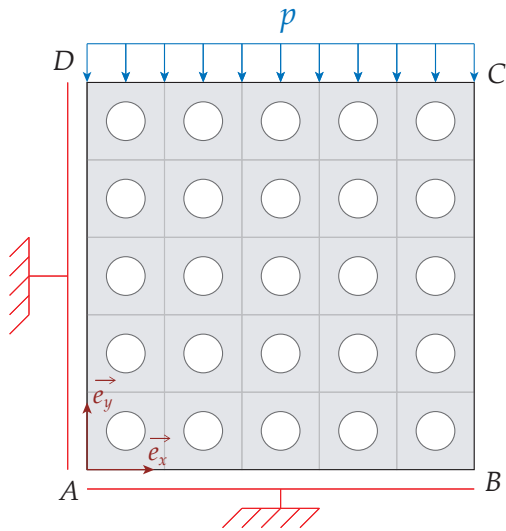


FIGURE 9.2 – SY-GL1 : Simulation d'un demi-problème

En utilisant la symétrie du problème, on travaille sur une demi-structure carrée avec les conditions aux limites représentées sur la Fig. 9.2. Les déplacements du bord gauche sont bloqués dans la direction \vec{e}_x et les déplacements du bord inférieur sont bloqués dans la direction \vec{e}_y .

Questions

1. Renommer le *Model SY-GL1*.
2. Dans le module *Part* :
 - (a) Créer un *Part* de type *2D-Planar, Deformable, et Shell* nommé *Part-Motif*.
 - (b) Définir la géométrie du motif dans le *Sketch*, comme cela est représenté sur la Fig. 9.3. On fera attention à ce que le point sur le cercle se trouve sur l'axe de symétrie horizontal ou vertical.

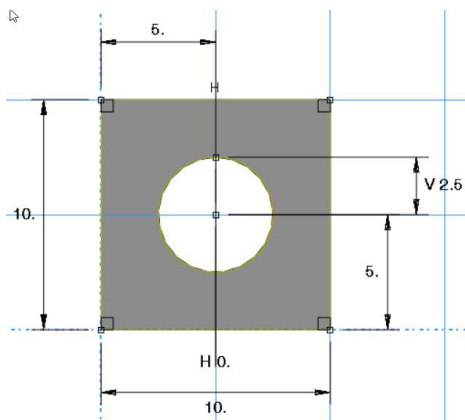


FIGURE 9.3 – Création de l'esquisse (*Sketch*)

- (c) Créer une *Partition* de type *Face → Sketch*. On décomposera le motif en huit parties. Pour cela, on tracera

des lignes passant par le centre du motif et orientées dans les directions horizontales, verticales, et selon les diagonales du carré. Le résultat devrait ressembler à celui représenté sur la Fig. 9.4.

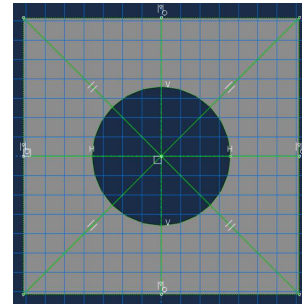


FIGURE 9.4 – Création de l'esquisse (*Sketch*)

3. Dans le module *Mesh* :
 - (a) Spécifier une densité globale de 2 mm.
 - (b) Spécifier une mailleur de type triangle non structuré.
 - (c) Choisir le type d'éléments triangulaire en choisissant l'option *Plane Stress*. Le maillage obtenu devrait ressembler à celui de la Fig. 9.5.

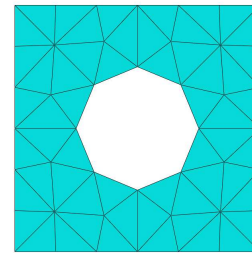


FIGURE 9.5 – Maillage d'un motif

4. Dans le module *Property* :
 - (a) Définir le *Material* de type *Mechanical → Elasticity*.
 - (b) Dans *Sections*, créer une *Section* de type *homogeneous*, l'épaisseur étant égale à 1 mm.

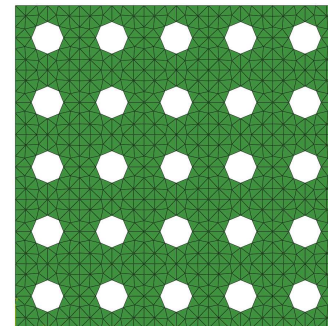


FIGURE 9.6 – Maillage

5. Dans le module *Assembly* :
 - (a) Créer une instance du motif de type *Dependent (mesh on Part)*.

- (b) Utiliser l'outil *Linear Pattern* pour créer un pavage de 5x5 motifs comme cela est représenté sur la Fig. 9.7. Construire un maillage global de la demi-structure en effectuant un assemblage (module *Assembly*) du maillage du motif.

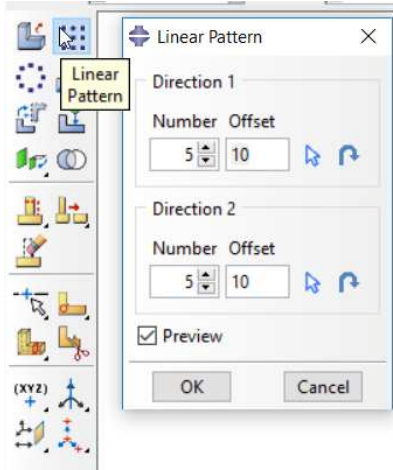


FIGURE 9.7 – Utilisation de l'outil *Linear Pattern*

- (c) Éliminer les nœuds doubles aux interfaces entre les motifs en utilisant l'outil *Merge/Cut instances* du module *Assembly*. Choisir *Operations* → *Merge* et la sous-option *Mesh*, comme cela est représenté sur la Fig. 9.8. Cliquer sur le bouton *Continue* puis sélectionner les instances à fusionner en cliquant sur le bouton *Instance Selection* en bas à droite de l'espace de travail. Le fait d'utiliser *Merge* a créé un nouveau *Part*.

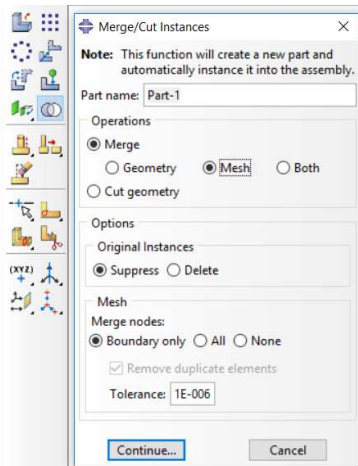


FIGURE 9.8 – Utilisation de la fonction *Merge* pour fusionner les nœuds superposés

- 6. Dans le module *Part* :
 - (a) Renommer *Part-Global* le *Part* ayant été créé.
 - (b) Créer un *Section Assignment* pour lier le nouveau *Part* au *Material* ayant été défini.
- 7. Dans le module *Step* :

- (a) Créer le *Step* de type *Static, General* avec un seul pas de temps.

- 8. Dans le module *Loads* :
 - (a) Imposer les conditions aux limites en déplacements (*BC*).
 - (b) Imposer les conditions aux limites en efforts (*load*). On pourra soit créer un *Load* de type *Pressure* et spécifier une valeur positive, soit créer un *load* de type *Surface traction* et imposer une valeur négative.
- 9. Dans le module *Job* :
 - (a) Créer un *Job* nommé *Job-SY-GL1*.
 - (b) Lancer un calcul.

10. Qu'est-ce qu'un effet de bord ? Où s'observe-t-il pour le cas test étudié ?

Le résultat obtenu devrait ressembler à celui représenté sur la Fig. 9.9.

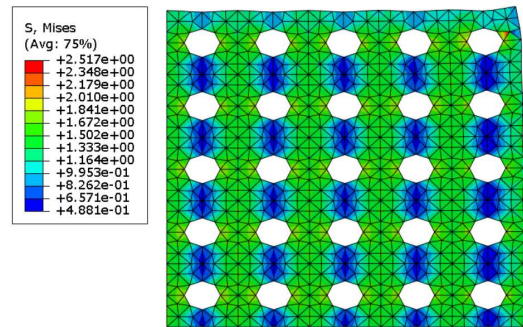


FIGURE 9.9 – SY-GL1 : Contrainte de Von Mises

1.3 Calcul global 2 (SY-GL2)

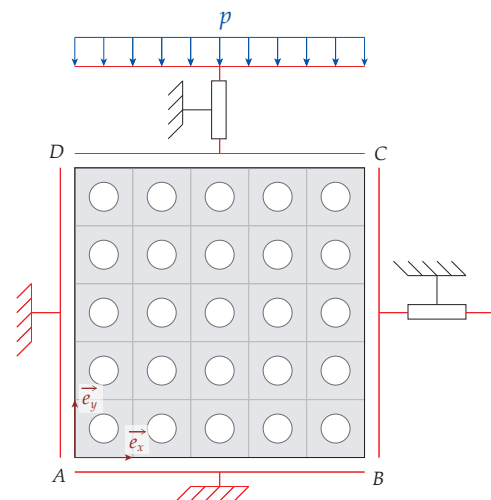


FIGURE 9.10 – SY-GL2 : Simulation d'un demi-problème avec les conditions aux limites modifiées

Pour le Test SY-GL2, on modifie les conditions aux limites pour empêcher les effets de bords, comme cela est représenté sur la Fig. 9.10. Pour imposer ces nouvelles conditions

aux limites on devra définir dans l'assemblage un *Reference Point*. Le déplacement vertical (selon \vec{e}_y) du *Reference Point* sera imposé égal au déplacement vertical du bord supérieur. Le déplacement horizontal (selon \vec{e}_x) du *Reference Point* sera imposé égal au déplacement horizontal du bord droit.

Questions

1. Dupliquer le *Model* et renommer le nouveau modèle *Model-SY-GL2*.
2. Dans le module *Assembly* :
 - (a) Depuis le menu déroulant supérieur *Tools* → *Reference Point*, créer un *Reference Point* qui permettra d'imposer les conditions additionnelles sur les bords hauts et droits du motif. Le *Reference Point* ne doit pas être un nœud du maillage existant, il est donc défini par ses coordonnées en dehors du domaine maillé. Par exemple, on peut choisir le point de coordonnées (55, 55) les dimensions étant données en mm.
 - (b) Créer un *Set* de type *Geometry* nommé *Set-RP* pour la *Reference Point*.
 - (c) Créer un *Set* de type *Node* nommé *Set-BC* pour le segment $[BC]$ et un autre nommé *Set-CD* pour le segment $[CD]$.
3. Dans le module *Interaction* :
 - (a) Utiliser l'outil *Create Constraint* pour créer une nouvelle *Constraint* de type *Equation*.
 - (b) Observé le tableau qui s'affiche et qui devrait ressembler à celui représenté sur la Fig. 9.11.

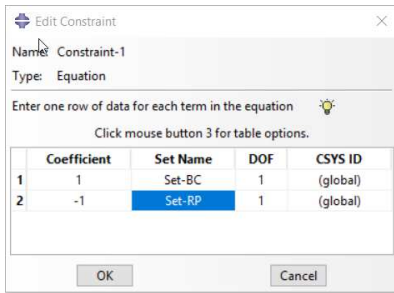


FIGURE 9.11 – Fenêtre permettant de définir une *Equation*

L'équation prise en compte par *Abaqus* est la somme de la contribution de chaque ligne du tableau, la contribution d'une ligne étant définie comme suit :

- La colonne *Coefficient* indique un coefficient multiplicateur de l'équation.
- La colonne *Set Name* indique le *Set* sur lequel porte la condition. Lorsque l'un des *Set* ne contient qu'un *Reference Point* celui-ci doit être placé sur la dernière ligne du tableau.
- La colonne *DOF* indique le numéro de la composante du déplacement sur laquelle porte l'équation. (1 → \vec{e}_x , 2 → \vec{e}_y , 3 → \vec{e}_z).

— La colonne *CSYS ID*, indique le repère dans lequel on se place, par défaut le repère global (\vec{e}_x , \vec{e}_y , \vec{e}_z).

4. Dans le module *Job* :
 - (a) Créer un *Job* nommé *Job-SY-GL2*.
 - (b) Lancer le calcul.
5. Que dire de l'effet de bord ?
6. Remplir la ligne SY-GL2 du Tableau 9.1 à partir des équations suivantes définissant la déformation moyenne $\hat{\epsilon}_{yy}$ et le module d'Young moyen \hat{E}_{yy} :

$$\hat{\epsilon}_{yy} = \frac{U_2(C)}{L_2},$$

$$\hat{E}_{yy} = \frac{\hat{\sigma}_{yy}}{\hat{\epsilon}_{yy}}.$$

	$\hat{\epsilon}_{yy}$	\hat{E}_{yy} (MPa)
SY-GL2	-6.941×10^{-6}	144076
SY-MO-RP	-6.941×10^{-6}	144076

TABLE 9.1 – Grandeurs homogénéisées pour les différents tests

Le résultat obtenu devrait ressembler à celui représenté sur la Fig. 9.12.

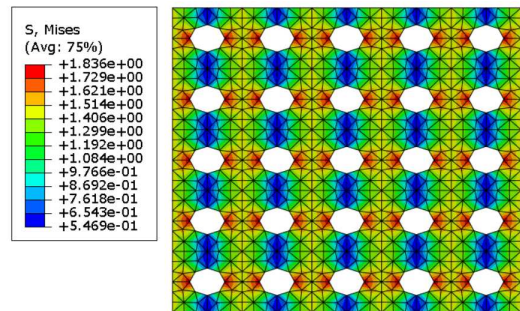


FIGURE 9.12 – SY-GL2 : Contrainte de Von Mises

1.4 Calcul sur 1 motif : méthode avec *Reference point*

On cherche maintenant à simuler la déformation d'un motif seulement. Dans un premier temps, on applique au motif les conditions aux limites représentées sur la Fig. 9.13. Les déplacements du bord inférieur et du bord gauche sont toujours bloqués selon \vec{e}_y et \vec{e}_x , respectivement.

Le résultat obtenu devrait ressembler à celui représenté sur la Fig. 9.14.

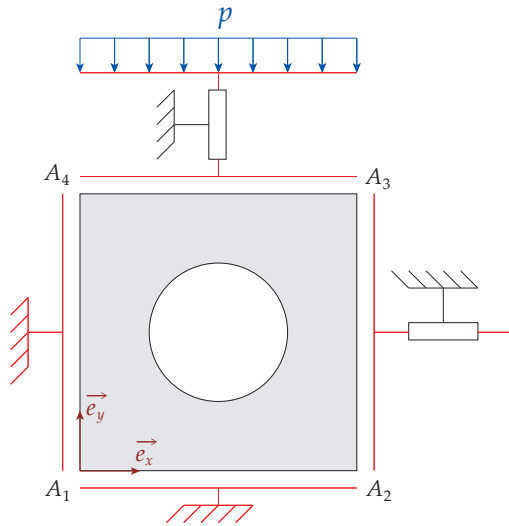


FIGURE 9.13 – SY-MO-RP : Simulation d'un motif de la structure

2 Motif asymétrique

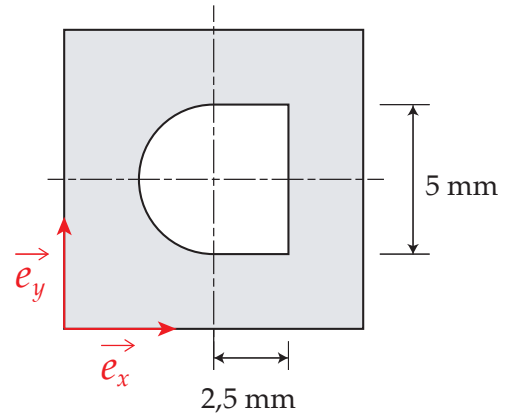


FIGURE 9.15 – Un motif alternatif de forme plus complexe

On considère maintenant un trou de forme un peu plus complexe représenté sur la Fig. 10.1.

Questions

1. Dupliquer le *Model SY-GL2*. Renommer le modèle *SY-MO-RP*.
2. Effectuer une simulation numérique pour un seul motif en appliquant les conditions aux limites représentées sur la Fig. 9.13.
3. À partir des résultats numériques, remplir la ligne *SY-MO-RP* du Tableau 9.1.

1. Réaliser un modèle *AS-GL* en définissant une structure complète 5x5 et en se basant sur les conditions aux limites de la Fig. 9.10.
2. Remplir la ligne *AS-GL* du Tableau 9.2.
3. Réaliser un modèle *AS-MO* en comportant un seul motif en se basant sur les conditions aux limites de la Fig. 9.13.
4. Remplir la ligne *AS-MO* du Tableau 9.2.

	$\hat{\epsilon}_{yy}$	\hat{E}_{yy} (MPa)
AS-GL2	-7.4554×10^{-6}	134 163.7
AS-MO-RP	-7.452×10^{-6}	134 194.1

TABLE 9.2 – Grandeurs homogénéisées pour les différents tests

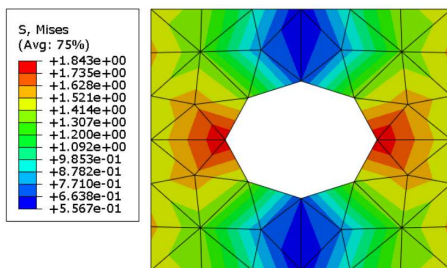


FIGURE 9.14 – SY-MO-RP : : Contrainte de Von Mises

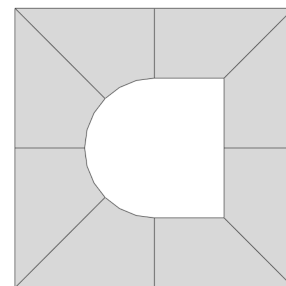


FIGURE 9.16 – AS-MO : Maillage et partitions

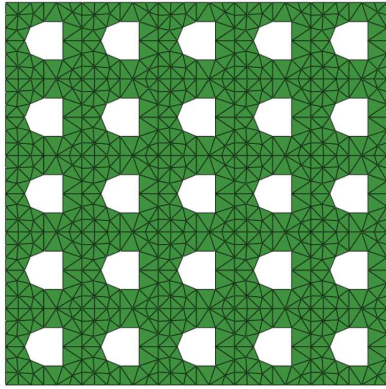


FIGURE 9.17 – AS-GL : Maillage

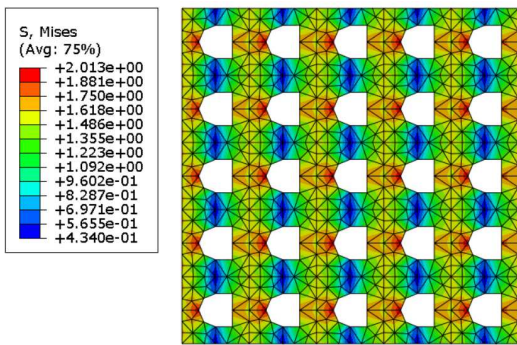


FIGURE 9.18 – AS-GL : Contrainte de Von Mises

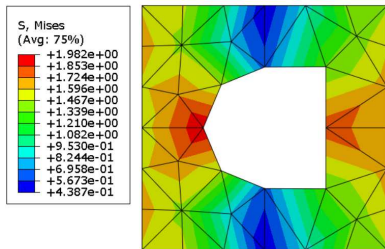


FIGURE 9.19 – AS-MO : Contrainte de Von Mises

Colloque National en Calcul des Structures CSMA. 9-13
Mai 2011. Presqu'île de Giens (Var).

Bibliographie

- [1] S. Lejeunes, S. Bourgeois. Une toolbox Abaqus pour le calcul de propriétés effectives de milieux hétérogènes. 10^e

TP2 : Utilisation de *Homtools*

1 Calcul numérique dans *Abaqus*

1.1 Calcul sur 1 motif : KUBC (*Homtools*)

1. Effectuer une simulation numérique pour un motif en utilisant l'outil KUBC de *Homtools*.

- Créer deux *Reference Points* pour appliquer le chargement.
- Créer un *Set* pour chaque *Reference Point*.
- Dans le module *Interaction*, à l'aide de la toolbox *Homtools* accessible depuis le menu *Plug-ins* → *Homtools* créer les *Constraints* à appliquer au motif.
- Appliquer les forces adéquates sur les *Reference Points* pour représenter la contrainte équivalente $\hat{\sigma}$ avec :

$$\hat{\sigma} = \begin{pmatrix} 0 & 0 \\ 0 & -p \end{pmatrix}.$$

2. Remplir la ligne SY-MO-KUBC du Tableau 9.1.

1.2 Calcul sur 1 motif : SUBC (*Homtools*)

1. Effectuer une simulation numérique pour un motif en utilisant l'outil SUBC de *Homtools*.
2. Remplir la ligne SY-MO-SUBC du Tableau 9.1.

1.3 Calcul sur 1 motif : PBC (*Homtools*)

1. Effectuer une simulation numérique pour un motif en utilisant l'outil PBC de *Homtools*.
2. Remplir la ligne SY-MO-PBC du Tableau 9.1.

2 Dalle percée d'un motif asymétrique

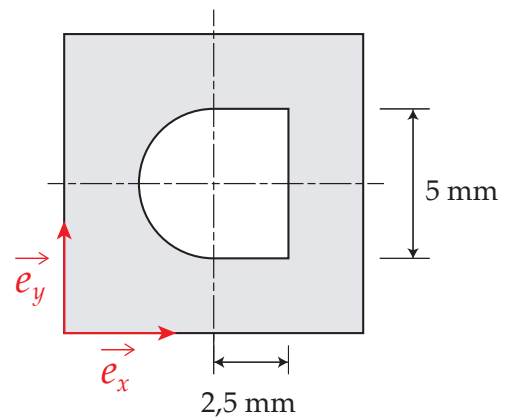


FIGURE 10.1 – Un motif alternatif de forme plus complexe

On ne considère plus un trou de forme circulaire, mais un trou de forme complexe représenté sur la Fig. 10.1. Reprendre les tests précédents avec la nouvelle géométrie asymétrique et remplir le Tableau 10.1.

	$\hat{\epsilon}_{yy}$	\hat{E}_{yy}
AS-GL2		
AS-MO-RP		
AS-MO-KUBC		
AS-MO-SUBC		
AS-MO-PBC		

TABLE 10.1 – Grandeurs homogénéisées pour les différents tests

ABAQUS

Plasticité

1 Plasticité

σ_{true} et ε_{ln}^{pl} en utilisant la définition est donnée dans la documentation *Abaqus* et rappelée ci-après :

$$\begin{aligned}\sigma_{true} &= \sigma (1 + \varepsilon), \\ \varepsilon_{ln}^{pl} &= \ln(1 + \varepsilon) - \frac{\sigma_{true}}{E}.\end{aligned}$$

2 Endommagement

Le taux de triaxialité L'endommagement des métaux est guidé en outre par le taux de triaxialité η défini par :

$$\eta = -\frac{p}{\sigma_{vm}},$$

p représentant la pression et σ_{vm} la contrainte de Von Mises définis par :

$$\begin{aligned}p &= -\frac{1}{3} \text{tr}(\boldsymbol{\sigma}), \\ \sigma_{vm} &= \sqrt{\frac{3}{2} \mathbf{s} : \mathbf{s}}, \\ \mathbf{s} &= \boldsymbol{\sigma} + p \mathbf{1},\end{aligned}$$

$\mathbf{1}$ représentant la matrice identité.

Initiation de l'endommagement Avec le *Ductile criterion*, l'initiation de l'endommagement est guidée par un scalaire ω_D défini sur le domaine Ω , l'endommagement s'initiant lorsque le critère suivant est vérifié :

$$\omega_D = 1.$$

ω_D est calculé à partir d'une variable scalaire nommée $\bar{\varepsilon}_D^{pl}$ en utilisant la formule suivante :

$$\omega_D = \int_t \frac{\dot{\varepsilon}^{pl}}{\bar{\varepsilon}_D^{pl}} dt.$$

En dérivant cette équation par rapport au temps, on obtient :

$$\dot{\omega}_D = \frac{\dot{\varepsilon}^{pl}}{\bar{\varepsilon}_D^{pl}}.$$

où $\bar{\varepsilon}_D^{pl}$ est une fonction de la dérivée temporelle $\dot{\varepsilon}^{pl}$ de la déformation plastique cumulée et du

Propagation de l'endommagement L'endommagement est caractérisé par un champ scalaire D variant de 0 (matériau sain) à 1 (matériau rompu) tel que :

$$\boldsymbol{\sigma} = (1 - D) \bar{\boldsymbol{\sigma}},$$

$\boldsymbol{\sigma}$ étant la matrice des contraintes et $\bar{\boldsymbol{\sigma}}$ la matrice des contraintes dites *effectives* correspondant aux contraintes que l'on aurait dans un matériau non endommagé. *Abaqus* propose différentes manières de calculer D à partir de \dot{u}^{pl} . Pour ce TP, on utilisera une évolution de l'endommagement basée sur :

- une évolution linéaire de l'endommagement D en fonction de \dot{u}^{pl} ,
- une valeur imposée de l'énergie de rupture par unité de surface notée G_f .

Avec ce choix d'options dans *Abaqus*, la loi d'évolution de la variable d'endommagement D est :

$$\dot{D} = \frac{L \sigma_{y0} \dot{\varepsilon}^{pl}}{2 G_f}.$$

Dans cette équation, σ_{y0} représente la valeur de la contrainte équivalente (*yield stress*), c'est à dire ici la contrainte de *Von Mises*, et L représente la longueur caractéristique d'un élément définie comme étant une approximation de la distance allant de part et d'autre de la bande endommagée.

Remarque : Pour que l'énergie de rupture soit bien en accord avec la valeur de G_f donnée, il faut que la contrainte équivalente ne change plus après initiation de l'endommagement, ce qui correspond à un comportement plastique parfait.

TP1 : Essai de traction

1 Données du problème

Géométrie On cherche à modéliser un essai de traction dont le montage est représenté sur les Fig. 12.1a et 12.1b.

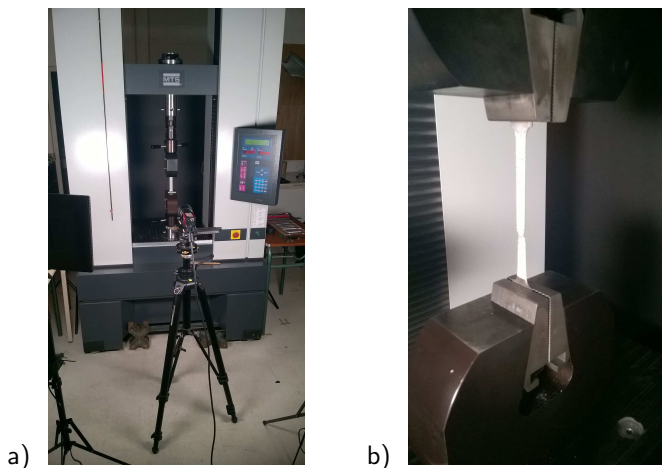


FIGURE 12.1 – Montage de l'essai de traction étudié

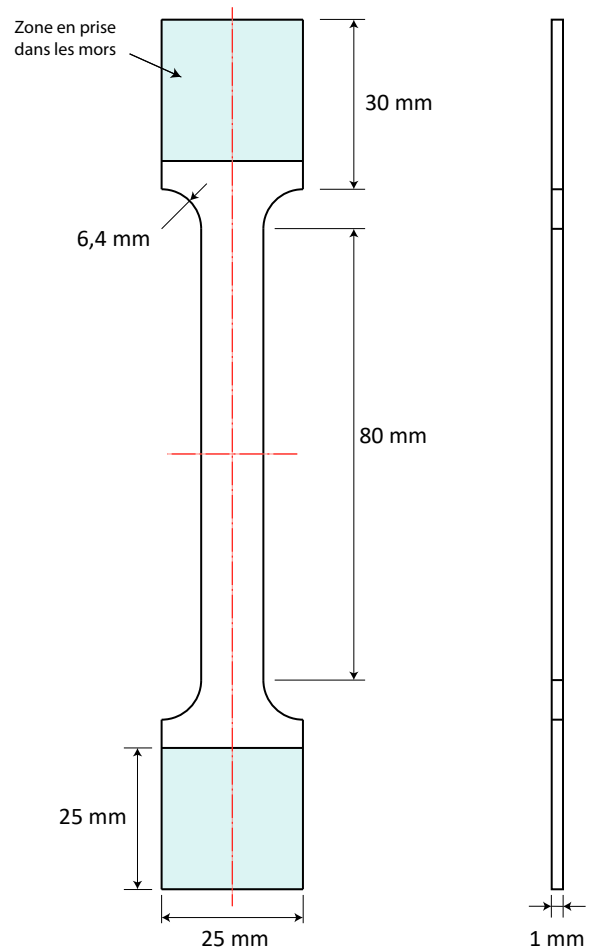


FIGURE 12.2 – Géométrie de l'éprouvette

La géométrie de l'éprouvette utilisée est représentée sur la Fig. 12.2. L'épaisseur de l'éprouvette est notée e avec :

$$e = 1 \text{ mm.}$$

Numériquement, les deux parties de l'éprouvette en prise dans les mors ne sont pas modélisées. L'étude se réduit donc à la géométrie représentée sur la Fig. 12.3a.

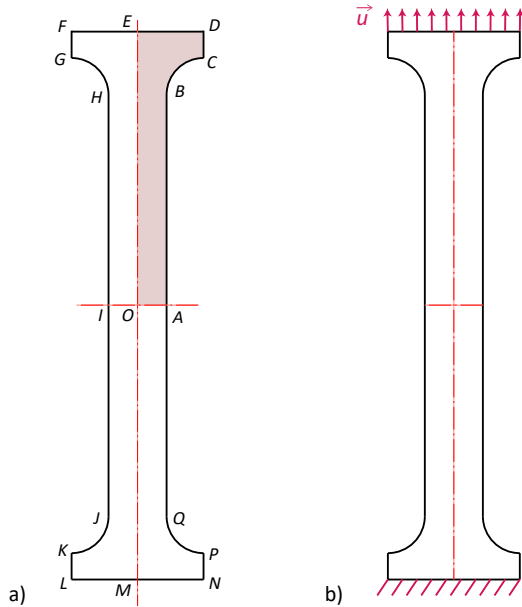


FIGURE 12.3 – a) Zone modélisée pour le calcul et b) conditions aux limites à appliquer

Matériau Le module d'Young du matériau est noté E et le coefficient de Poisson est noté ν . Dans un premier temps, on considère que ces deux grandeurs prennent les valeurs numériques suivantes (affinera ensuite ces valeurs à l'aide des résultats expérimentaux) :

$$E = 80 \text{ GPa},$$

$$\nu = 0.3.$$

Conditions aux limites Les conditions aux limites représentées sur la Fig. 12.3b. Le long du segment $[LM]$ l'éprouvette est en liaison encastrement dans le mors du bas supposé rigide :

$$u_x = 0,$$

$$u_y = 0.$$

Le long du segment $[DF]$, les conditions aux limites s'écrivent :

$$u_x = 0,$$

$$u_y = u_1.$$

La valeur de u_1 sera donnée séparément pour chaque partie de l'exercice.

2 Résultats expérimentaux

Des mesures sont réalisées pendant toute la durée de l'essai mécanique :

- Le déplacement de la traverse est mesuré en continu pendant l'essai ce qui permet d'extraire une courbe $u_1(t)$ représentant le déplacement vertical du segment $[DF]$.

- Des mesures de force sont réalisées par la cellule de force de la machine. Ces mesures permettent d'extraire la courbe $\mathcal{F}_y(t)$ indiquant la force exercée par le mors du haut sur l'éprouvette au niveau du segment $[DF]$.

Les valeurs de $u_1(t)$ et de $\mathcal{F}_y(t)$ sont données sous forme d'un ensemble de points listés dans le fichier `resultats_essais.txt` pouvant être téléchargé depuis l'ENT puis importé dans un tableur. Les courbes $\mathcal{F}_y = f(u_1)$ obtenues sont représentées sur la Fig. 12.4 (en se focalisant sur la partie élastique de la courbe et le début de la plasticité) et la Fig. 12.5 (en affichant la courbe complète).

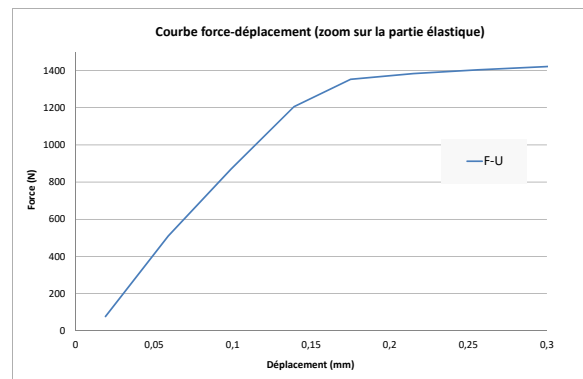


FIGURE 12.4 – Courbe force-déplacement en se focalisant sur la partie élastique de la courbe et le début de la plasticité

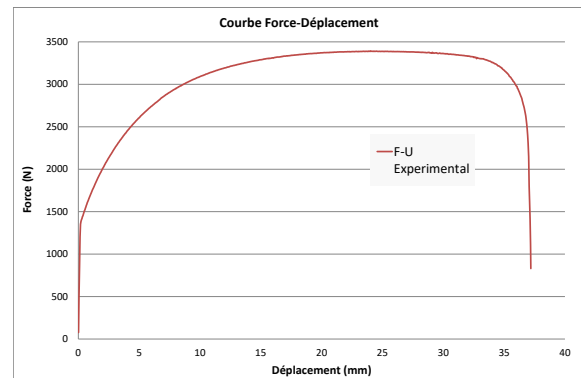


FIGURE 12.5 – Courbe force-déplacement globale

Questions

1. Dans un tableur, importer les données du fichier `resultats_essais.txt`.
2. Dans un tableur, tracer la loi globale $\mathcal{F}_y = f(u_1)$ caractéristique de l'essai.

3 Travaux à effectuer dans Abaqus

3.1 Premier calcul élastique

Questions

1. Renommer le modèle *Model-Elast*.
2. Dans le module *Part* :
 - (a) Créer un *Part* pour l'éprouvette en sélectionnant les options *2D* et *Deformable*.
 - Dans l'esquisse (*Sketch*), créer un point O de coordonnées $(0, 0)$. Appliquer un contrainte de type *Fixed* sur ce point.
 - Créer les 2 axes de symétrie horizontaux et verticaux de l'éprouvette passant par O en utilisant l'outil *Construction Lines*.
 - Créer la géométrie du quart supérieur droit de l'éprouvette reliant les points $OABCDE$. Placer l'intégralité des cotes de telle sorte que l'éprouvette ne puisse pas être déformée avec la souris et l'outil *Drag Entities*.
 - Créer la géométrie de l'éprouvette entière en utilisant l'outil *Mirror* → *Copy*.
 - (b) Créer une partition de type *Face* → *Sketch* pour introduire dans la géométrie les segments $[IO]$ et $[OA]$.
 - (c) Créer les Sets suivants :
 - Créer un Set nommé *Set-Top* pour le segment $[DF]$ afin d'extraire plus facilement la résultante des efforts extérieurs sur ce segment.
 - Créer un Set nommé *Set-E* pour le point E afin d'extraire plus facilement le déplacement selon \vec{e}_y en ce point.
 - Créer un Set nommé *Set-center* pour le point O afin d'extraire plus facilement la valeur de la déformation et de la contrainte en ce point.
3. Dans le module *Mesh* :
 - (a) Utiliser l'outil *Assign Mesh Controls* pour sélectionner les options *Element Shape* → *Quad*, *Technique* → *Free* et *Algorithm* → *Medial Axis*.
 - (b) Utiliser l'outil *Assign Element Type* pour sélectionner les options *Family* → *Plane Stress*, *Geometric Order* → *Linear*. Décocher la case *Reduced integration*.
 - (c) Utiliser l'outil *seed Part* pour imposer une densité globale de 1.5 mm pour chaque élément.
 - (d) Utiliser l'outil *Seed edges* pour imposer un nombre de 5 éléments sur les segments $[IO]$ et $[OA]$.
4. Dans le module *Property* :
 - (a) Définir le matériau élastique.
 - (b) Définir une *Section* de type *Homogeneous*.
 - (c) Effectuer le *Section assignment* pour relier le matériau et la géométrie.
5. Dans le module *Assembly* :
 - (a) Créer une instance de l'éprouvette de type *Dependent*.
6. Dans le module *Step* :
 - (a) Créer un *Step* pour un matériau élastique.
 - (b) Créer un *History Output Requests* nommé *H-output-F*. Choisir *Domain* → *Set* et sélectionner *Set-Top*. Sélectionner *Forces/Reactions* → *RF*, *Reaction forces and moments* → *RF2*.
 - (c) Créer un *History Output Requests* nommé *H-output-U*. Choisir *Domain* → *Set* et sélectionner *Set-E*. Sélectionner *Displacement/velocity/Acceleration* → *T*, *translations and rotations* → *U2*.
7. Dans le module *Load* :
 - (a) Appliquer les conditions aux limites appropriées le long du segment $[LM]$.
 - (b) Appliquer les conditions aux limites appropriées le long du segment $[FD]$ avec $u_1 = 1$ mm.
8. Dans le module *Job* :
 - (a) Créer un *Job*.
 - (b) Lancer un calcul.
9. Dans le module *Visualization* :
 - (a) Afficher la contrainte de Von Mises.

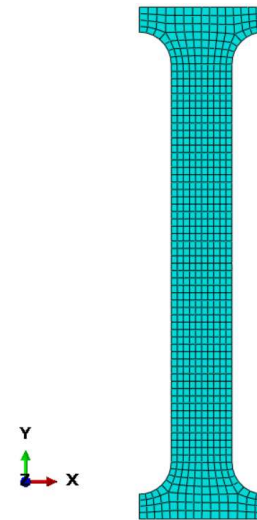


FIGURE 12.6 – Maillage

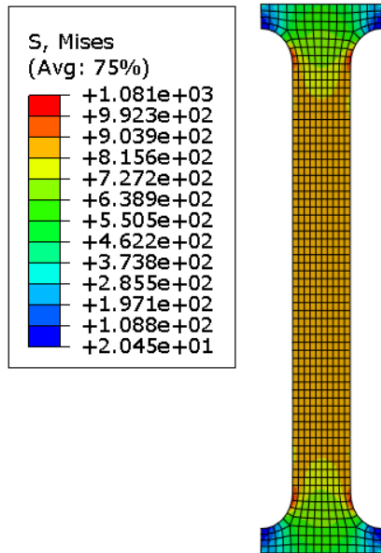


FIGURE 12.7 – Contrainte de Von Mises

- (b) Extraire la résultante \mathcal{F}_y des efforts appliqués sur le segment $[DF]$. Pour cela, aller dans *ODB history output*, sélectionner l'ensemble des forces *RF2*, cliquer sur *Save As* et choisir *Save Operation* → *Sum*.

$$\mathcal{F}_y = 10019.7 \text{ N}$$

- (c) Extraire la déformation ε_{yy} et la contrainte σ_{yy} au point O .

— Utiliser l'outil *Create XY Data*, choisir *ODB field output*. Dans l'onglet *Variables*, encart *Output Variables*, choisir *Position* → *Unique Nodal* et cocher les cases *E : Strain components* → *E22* et *S : Stress components* → *S22*. Dans l'onglet *Elements/Nodes*, encart *Selection*, sélectionner *Method* → *Node sets* et *PART-1-1.SET-CENTER*. Cliquer sur *Save*.

— Deux nouvelles lignes apparaissent dans l'arborescence de gauche sous *XYData* (une pour la déformation et une pour la contrainte). Faire *Clic-Droit* → *Edit* pour relever la valeur de ε_{yy} et σ_{yy} .

$$\begin{aligned} \varepsilon_{yy} &= 0.01026, \\ \sigma_{yy} &= 821.289 \text{ MPa}. \end{aligned}$$

- (d) Calculer les ratios α et β à partir du calcul élastique réalisé :

$$\begin{aligned} \alpha &= \frac{\varepsilon_{yy}}{u_1}, \\ \beta &= \frac{\sigma_{yy}}{\mathcal{F}_y}. \end{aligned}$$

$$\begin{aligned} \alpha &= 0.01026 \text{ mm}^{-1}, \\ \beta &= 0.08197 \text{ mm}^{-2}. \end{aligned}$$

- (e) Vérifier que les valeurs obtenues pour α et β sont proches des valeurs approximatives suivantes :

$$\begin{aligned} \alpha &= \frac{1}{\|ND\|}, \\ \beta &= \frac{1}{e \|IA\|}. \end{aligned}$$

$$\begin{aligned} \alpha &= 0.009728 \text{ mm}^{-1}, \\ \beta &= 0.081967 \text{ mm}^{-2}. \end{aligned}$$

- (f) Calculer le taux de triaxialité au niveau du point central de l'éprouvette.

Le taux de triaxialité η est défini par :

$$\eta = -\frac{p}{\sigma_{vm}}$$

Les résultats numériques donnent :

$$\begin{aligned} p &= -273.763 \text{ MPa}, \\ \sigma_{vm} &= 821.289 \text{ MPa}, \end{aligned}$$

ce qui donne :

$$\eta = 0.3333.$$

10. Dans le tableau :

- (a) Utiliser les paramètres α et β pour calculer une colonne pour la déformation ε_{yy} et une autre pour la contrainte σ_{yy} .

- (b) Sur une feuille de calcul intitulée *Courbe sig-eps*, tracer loi locale $\sigma_{yy} = f(\varepsilon_{yy})$ caractéristique du matériau.

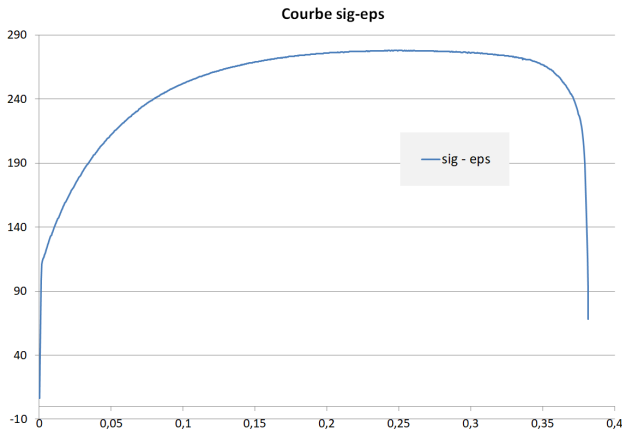


FIGURE 12.8 – Courbe contrainte-déformation

- (c) Dupliquer la courbe montrant la loi de comportement $\sigma_{yy} = f(\varepsilon_{yy})$ sur une feuille de calcul intitulée *Courbe sig-eps (zoom)*. Faire un zoom sur la partie élastique de la courbe. Pour cela double-cliquer sur la graduation des axes et changer les extremums.

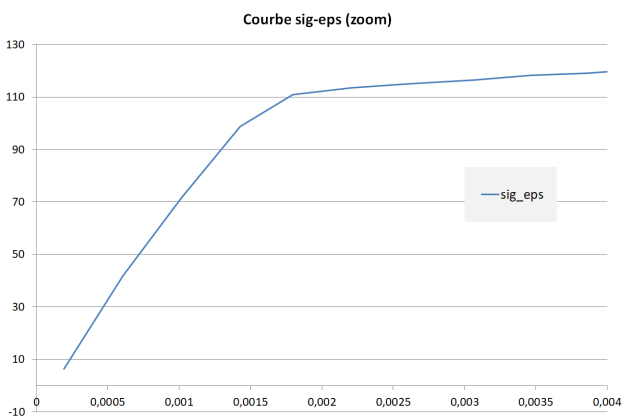


FIGURE 12.9 – Courbe contrainte-déformation (zoom)

- (d) En se basant sur la partie linéaire de la courbe, déduire une valeur corrigée du module d'Young E du matériau. On pourra appliquer la formule :

$$E = \frac{\sigma_2 - \sigma_1}{\varepsilon_2 - \varepsilon_1}$$

$$E = \frac{98.852714 - 6.3114994}{0.00142614 - 0.00019494}$$

$$= 75163,436$$

4 Modélisation de la zone plastique

4.1 Model-Plast1 : début de la courbe

Dans cette section, on cherche à modéliser le début de la courbe Force-déplacement représenté sur la Fig. 12.4.

Questions

- Dans le tableur utilisé, calculer σ_{true} et ε_{ln}^{pl} en utilisant la définition est donnée dans la documentation *Abaqus* et rappelée ci-après :

$$\sigma_{true} = \sigma (1 + \varepsilon),$$

$$\varepsilon_{ln}^{pl} = \ln(1 + \varepsilon) - \frac{\sigma_{true}}{E}.$$

- Dupliquer le modèle *Model-Elast* pour créer un nouveau modèle nommé *Model-Plast* servant de base aux calculs prenant en compte la plasticité.
- Dans le module *Material* :

- Corriger la valeur donnée au module d'Young du matériau à partir des résultats de la partie précédente.

À partir de cette question, on considère que :

$$E = 75163 \text{ MPa.}$$

- Rajouter dans le comportement du matériau une partie de type *Plastic* avec l'option *Hardening* → *Isotropic*. Dans l'encart *Data* insérer 7-8 lignes de valeurs pour caractériser la partie initiale de l'essai (Fig. 12.4). La colonne *Yield Stress* doit contenir les valeurs de σ_{true} et la colonne *Plastic Strain* doit contenir les valeurs de ε_{ln}^{pl} . On pourra par exemple se baser sur les mesures aux temps 32.4, 33.8, 38.6, 45, 51.4, 57.8, et 65.8 s. On remplacera la première valeur de la colonne *Plastic* pour éviter de voir un message d'erreur apparaître au moment de lancer le *Job*.

Data		
	Yield Stress	Plastic Strain
1	98.9934612434951	0
2	111.1010493	0.000315764
3	116.833002863075	.0015106010361136
4	122.619485428791	.0030299676070908
5	128.265417339093	.0045997067280873
6	133.927879973204	0.006156332211777
7	140.16168389436	.0081096225957552
8	150.479776944367	.011627154386194

FIGURE 12.10 – Définition du matériau plastique dans Abaqus

4. Dans le module *Step* :

- (a) Conserver dans *Incrementation* le *Type* : *Automatic*.
- (b) Spécifier un *Maximum number of increments* de 1000.
- (c) Spécifier un *Increment size* : *Initial* de 0.001.
- (d) Conserver un *Increment size* : *Minimum* de 10^{-5} .

5. Dans le module *Load* :

- (a) Se limiter à un déplacement u_1 maximal de 0.4 mm.

6. Lancer le *Job*.

7. Dans le module *Visualization* :

- (a) Afficher la déformation plastique cumulée ϵ^{pl} (*field* : *PEEQ*). D'après le *Analysis User's Guide* section *Materials* → *Inelastic Mechanical Properties* → *Classical metal plasticity*, ϵ^{pl} est défini par l'intégration sur le temps suivante :

$$\epsilon^{pl} = \epsilon^{pl} + \int_0^t \sqrt{\epsilon^{pl} : \epsilon^{pl}} dt.$$

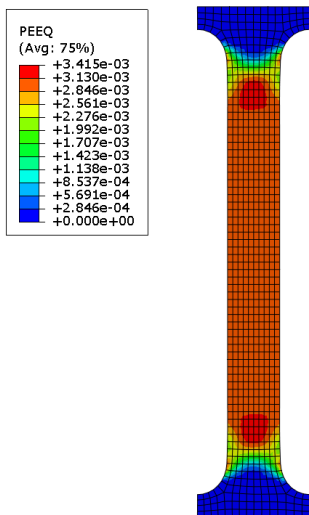


FIGURE 12.11 – Champ de la déformation plastique cumulée ϵ^{pl}

8. Dans un tableur, exporter la loi $\mathcal{F}_y = f(u_1)$. La tracer dans le même graphique que la courbe expérimentale. Vérifier la bonne superposition des deux courbes.

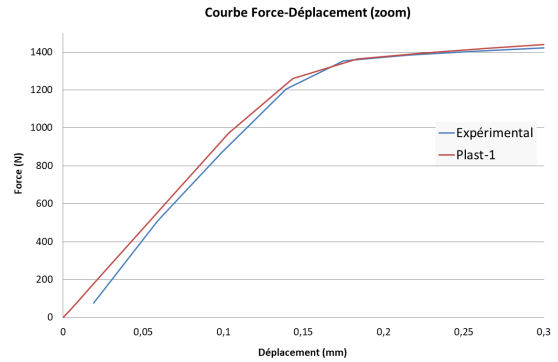


FIGURE 12.12 – Courbe de traction (zoom)

4.2 Model-Plast2 : ajout de la partie durcissante

Dans cette partie, on cherche à modéliser la partie durcissante (c'est à dire la partie croissante) de la courbe Force-déplacement représentée sur la Fig. 12.5.

Questions

1. Dupliquer le modèle *Model-Plast*. Renommer le nouveau modèle *Model-Plast2*.
2. Dans le module *Material*, ajouter des points dans la courbe de comportement du matériau. On pourra rajouter un point pour chaque nouvelle dizaine de mégapascal.
3. Dans le module *Loab*, modifier les conditions aux limites pour aller jusqu'à un déplacement maximal de 40 mm.
4. Dans le module *Job* lancer un nouveau *Job* nommé *Job-Plast2*.
 - Il se peut que le calcul plante. On ira alors dans le module *Step* et dans le menu déroulant supérieur on choisira *Other* → *General Solution Controls* → *Edit* → *Step1*. Une fenêtre de *Warning* s'affiche alors et on cliquera sur le bouton *Continue*. Dans la fenêtre *General Solution Controls Editor* qui s'affiche, on sélectionnera le bouton radio *Specify*. Dans l'onglet *Time Incrementation*, on spécifiera pour le paramètre I_A la valeur 50.
 - Pour ce calcul, il est important que les éléments du maillage soient des quadrangles. Avec des triangles, le calcul a tendance à planter pour un déplacement avoisinant 20 mm.
5. Tracer la courbe force-déplacement et la superposer dans le tableur à la courbe expérimentale.
6. Pour améliorer la superposition entre les courbes numériques et expérimentales :
 - Aller dans le module *Step* et éditer le *Step*.
 - Dans l'onglet *Basic*, mettre l'option *NIGeom* sur *On* pour activer les grands déplacements.
 - Relancer le calcul.

7. Tracer la courbe force-déplacement et la superposer dans le tableur à la courbe expérimentale et à la courbe obtenue avec l'hypothèse de petite perturbation.

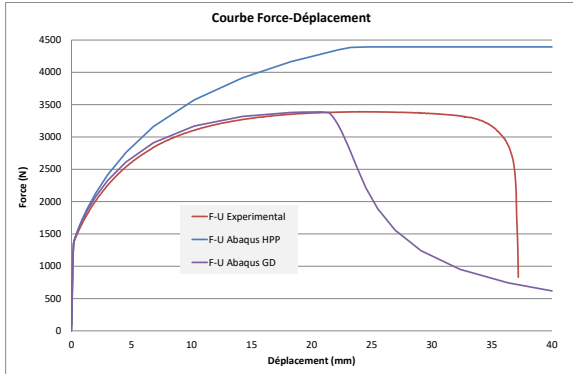


FIGURE 12.13 – Courbe de traction

3. Dans le tableur, ajouter la nouvelle courbe force-déplacement aux autres courbes déjà tracées

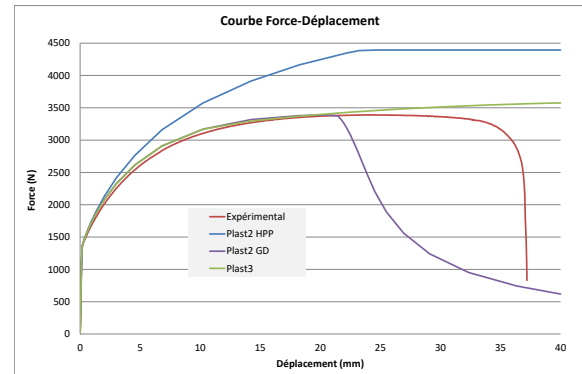


FIGURE 12.15 – Courbe de traction

4.3 Model-Plast3 : réduction du necking

1. Dupliquer le modèle *Model-Plast2*. Renommer le nouveau modèle *Model-Plast3*.
2. Dans le module *Material*, modifier la loi de comportement du matériau en la rendant plus durcissante pour éviter le phénomène de *necking* avant d'avoir atteint un déplacement de 32 ou 33 mm.

Data	Yield Stress	Plastic Strain
1	98.9934612434951	0
2	111.1010493	0.000315764
3	116.833002863075	0.00151060103611363
4	122.619485428791	0.00302996760709083
5	128.265417339093	0.00459970672808731
6	133.927879973204	0.006156332211777
7	140.16168389436	0.00810962259575521
8	150.479776944367	0.0116271543861945
9	160.36742173956	0.0151270127487487
10	170.225578495948	0.019026539202452
11	180.230020074114	0.0233085163590035
12	190.393759019569	0.0279775928725187
13	200.469374986604	0.0330115598336862
14	210.378053612068	0.038405735475769
15	220.128608868052	0.0441735515381369
16	230.160703461273	0.0506919333593313
17	240.141410733397	0.0579307983566327
18	250.249535600409	0.0654881016310587
19	260.164045330338	0.0741401793153216
20	270.161095990358	0.0838448802588919
21	280.178787331784	0.0949438512391215
22	290.230053587093	0.107389350036992
23	300.246022147908	0.121493023667771
24	310	0.14
25	410	0.34
26	510	0.54
27	610	0.74
28	710	0.94

FIGURE 12.14 – Nouvelles données utilisées pour la loi de comportement plastique

5 Ajout de l'endommagement

Il est possible d'aller plus loin dans la simulation du comportement du matériau, en modélisant non seulement la partie durcissante mais aussi la partie adoucissante de la courbe force-déplacement, *Abaqus* permet de prendre en compte l'endommagement du matériau. Pour utiliser l'endommagement dans *Abaqus*, il faut définir un critère d'*initiation* et une loi de *propagation* de l'endommagement. Un critère de suppression des éléments peut également être défini pour modéliser la rupture finale du matériau.

Questions

1. Dupliquer le modèle *Model-Plast3*. Renommer *Model-Plast-Endo* le nouveau modèle.
2. Dans le module *Property* :
 - (a) Ajouter de l'endommagement dans le modèle en allant dans *Material* et en sélectionnant *Mechanical* → *Damage For Ductile Metals* → *Ductile Damage*.
 - (b) Pour le paramètre *Fracture Strain*, choisir une valeur pas trop élevée comme $\bar{\epsilon}^{pl} = 0.30$.
 - (c) Pour le paramètre *Stress Triaxiality*, retourner dans le calcul élastique effectué au début du TP, et estimer le taux de triaxialité au centre de l'éprouvette.
 - (d) Pour *Strain Rate* choisir $\dot{\epsilon}^{pl} = 0$.
 - (e) Dans *Suboptions* → *Damage Evolution*, choisir *Type* : *Energy*, *Softening* : *Linear* et une *Fracture Energy* autour de $G_f = 300 \text{ N.mm}^{-2}$.

3. Dans le module *Step* :

(a) Utiliser l'outil *Create Field Output* pour créer un nouveau *Field Output Request* pour la variable d'endommagement D . Dans la fenêtre *Output Variables*, cocher le champ *Failure/fracture* → *SDEG*, *Scalar stiffness degradation*.

4. Dans le module *Job* :

- (a) Créer un *Job*.
- (b) Lancer un calcul.

5. Observer la répartition de la variable d'endommagement D au cours du calcul (*Field* : *SDEG*).

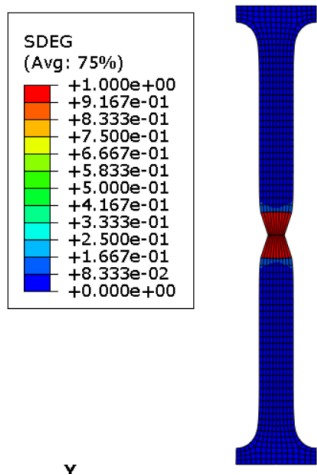


FIGURE 12.16 – Champ d'endommagement

- 6. Lancer le calcul.
- 7. Dans le tableur, exporter la courbe force-déplacement et la tracer dans un graphique en la superposant aux autres courbes Force-Déplacement déjà tracées.

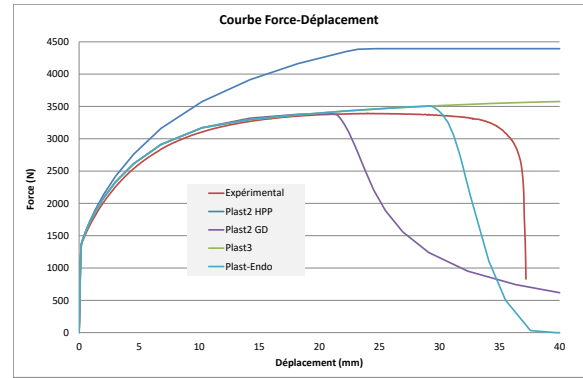


FIGURE 12.17 – Courbe de traction

6 Ajout de la rupture

Il est possible dans *Abaqus* de définir une valeur D_{max} de l'endommagement à partir de laquelle les éléments sont supprimés.

Questions

1. Dans le module *Mesh*, modifier la définition des éléments-finis pour imposer l'endommagement maximal $D_{max} = 0.99$.
2. Lancer à nouveau le calcul après avoir coché la case *State/Field/User/Time* → *STATUS*, *Status (some failure and plasticity models; VUMAT)* dans les *Field Output Request* Quelle différence observe-t-on dans l'affichage des résultats ?

ABAQUS

Mécanique de la rupture

Introduction

Objectifs L'objectif des trois séances de TP *Abaqus* sur la mécanique de la rupture est d'apprendre à utiliser les méthodes disponibles dans *Abaqus* pour simuler la fissuration des matériaux. Les résultats obtenus avec les différentes approches seront comparés entre eux à des solutions analytiques.

Les différents modèles Il existe différentes manières de modéliser la rupture, classifiées ci-dessous en 3 principaux types.

- Un *modèle de Griffith* peut être utilisé, cette approche étant également nommée *mécanique linéaire de la rupture*. Ceci signifie que les bords de la fissure sont libres d'efforts lorsqu'ils ne sont pas en contact. La propagation de la fissure est guidée par un critère énergétique et l'initiation d'une fissure n'est pas prédite *a priori*.
- Un *modèle cohésif* est défini si des efforts de cohésion persistent au voisinage de la pointe de la fissure, sous forme de tractions cohésives. L'initiation et la propagation de la fissure dépendent de l'état de contraintes au niveau de la pointe de la fissure.
- Un *modèle de fissuration diffuse* peut être défini pour éviter d'introduire une discontinuité dans le maillage pendant un calcul. On considère alors que la fissure possède une épaisseur. C'est alors une bande de matériau qui s'endommage au cours du calcul.

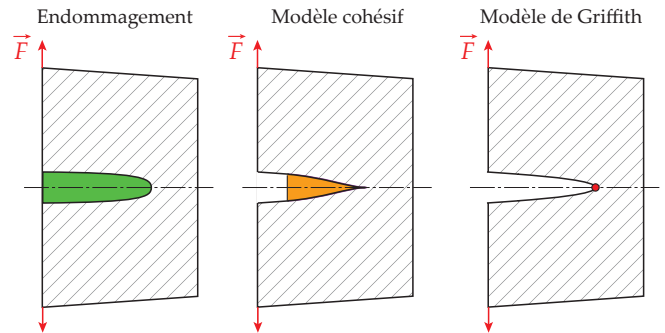


FIGURE 13.1 – Comparaison des modèles d'endommagement, cohésifs, et de Griffith

1 Modèles discontinus de rupture

1.1 Généralités

Prise en compte de la discontinuité La surface fissurée est notée Γ_c et son vecteur normal est noté \vec{n} (voir Fig. 13.2a). On peut noter que le sens de \vec{n} est arbitraire et permet de distinguer le bord supérieur de la fissure noté Γ_c^+ du bord inférieur noté Γ_c^- (voir Fig. 13.2b). On note $\llbracket u \rrbracket$ le saut de déplacement au niveau de la surface fissurée Γ_c défini par :

$$\llbracket u \rrbracket = \vec{u}^+ - \vec{u}^-,$$

\vec{u}^+ et \vec{u}^- désignant les déplacements sur les bords Γ_c^+ et Γ_c^- de la fissure, respectivement. On note \vec{t} la traction dans la fissure. L'équilibre au niveau de la fissure s'écrit :

$$\begin{aligned} \sigma^+ \cdot \vec{n} &= \vec{t}, & \text{sur } \Gamma_c^+, \\ \sigma^- \cdot \vec{n} &= \vec{t}, & \text{sur } \Gamma_c^-, \end{aligned}$$

σ^+ et σ^- représentant la contrainte sur les bords Γ_c^+ et Γ_c^- , respectivement. Pour prendre en compte un contact éventuel entre les deux bords de la fissure, la condition suivante est imposée :

$$\llbracket u \rrbracket \cdot \vec{n} > 0.$$

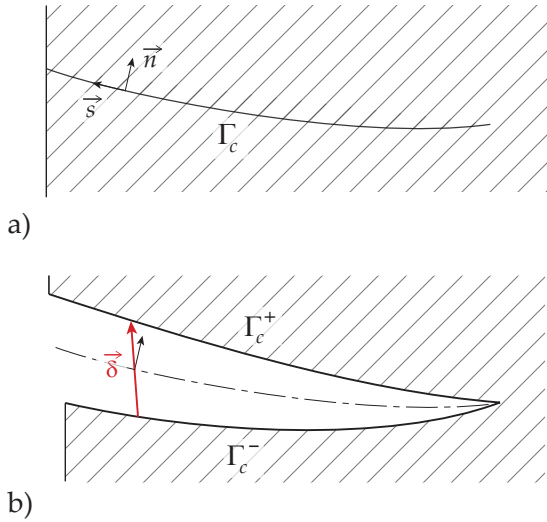


FIGURE 13.2 – a) Trajet de la fissure dans la configuration non déformée et b) ouverture de la fissure dans la configuration déformée

Critères d'orientation des fissures Ces critères sont uniquement utilisés lorsque le trajet de la fissure n'est pas connu *a priori*, c'est-à-dire dans *Abaqus* avec la méthode *X-Fem*. Nous utiliserons principalement le critère *maximum principal stress*.

1.2 Mécanique linéaire de la rupture

Traction dans la fissure Pour le modèle de Griffith, les bords de la fissure sont libres si il n'y a pas contact entre ses deux bords :

$$\text{Modèle de Griffith : } \vec{t} = \vec{0},$$

\vec{t} étant la traction dans la fissure. Tous les phénomènes de décohésion sont supposés être concentrés au niveau de la pointe de la fissure, c'est-à-dire un point en 2D et une courbe en 3D.

Propagation en mode I Le modèle de Griffith a comme seul paramètre l'énergie surfacique de rupture G_c . Le critère de propagation de la fissure s'écrit :

$$\text{Arrêt de la fissure : } G < G_c, \quad (13.2a)$$

$$\text{Propagation de la fissure : } G = G_c. \quad (13.2b)$$

Dans cette équation, G est le taux de restitution d'énergie élastique défini par :

$$G = \frac{d(\mathcal{W}_{ext} - \Psi)}{d\mathcal{A}}, \quad (13.3)$$

\mathcal{W}_{ext} étant le travail des efforts extérieurs, Ψ l'énergie libre du domaine considéré, et \mathcal{A} l'aire de la fissure. Dans le cas d'une propagation stationnaire de la fissure, G peut être estimé pour une propagation de fissure dans la direction \vec{u}_x par l'intégrale de contour J définie par :

$$J = \int_{\Gamma} (\psi \vec{u}_x - \sigma \cdot \frac{\partial \vec{u}}{\partial \vec{x}}) \cdot \vec{n} \, d\Gamma, \quad (13.4)$$

ψ étant l'énergie de déformation par unité de volume, Γ un contour d'intégration quelconque englobant la pointe de la fissure, et \vec{n} la normale extérieur à Γ . Le critère de propagation s'écrit alors $J = G_c$.

Propagation en modes mixtes Il existe différentes manières de généraliser la théorie de Griffith en modes mixtes. On détaille ici la *Power form law* qui sera utilisée pour les TP. Pour cette méthode, le critère de propagation 13.8b est réécrit sous la forme :

$$\frac{G}{G_c} = 1.$$

Avec la *Power form law*, ce rapport est calculé comme suit :

$$\frac{G}{G_c} = \left(\frac{G_n}{G_c^n}\right)^\alpha + \left(\frac{G_s}{G_c^s}\right)^\alpha + \left(\frac{G_t}{G_c^t}\right)^\alpha.$$

Dans la pratique on utilisera dans les TP des paramètres vérifiant :

$$G_c^m = G_c^s = G_c^t, \\ \alpha = 1.$$

Les autres lois disponibles dans *Abaqus* sont nommées *Benzeggagh-Kenane (BK) law* [6] et *Reeder law* [5]. Elles sont documentées dans l' *Analysis User's Guide* [2], section 10.7.1 → Applying the VCCT technique to the XFEM-based LEM approach → Mixed mode behavior.

1.3 Modèle cohésif

Définition Pour le modèle cohésif, la force surfacique de traction t est non nulle au voisinage de la pointe de la fissure et est calculée en fonction du saut de déplacement $\llbracket u \rrbracket$ dans la fissure. Pour cela, une loi cohésive reliant une traction équivalente t_{eq} à un saut de déplacement équivalent $\llbracket u \rrbracket_{eq}$ est définie, comme par exemple la loi représentée sur la Fig. 13.3. Les variables équivalentes t_{eq} et $\llbracket u \rrbracket_{eq}$ sont souvent définies de telle sorte que l'aire sous la loi cohésive est égale à l'énergie de fissuration par unité de surface G_c dans le cas d'une propagation stationnaire de la fissure.

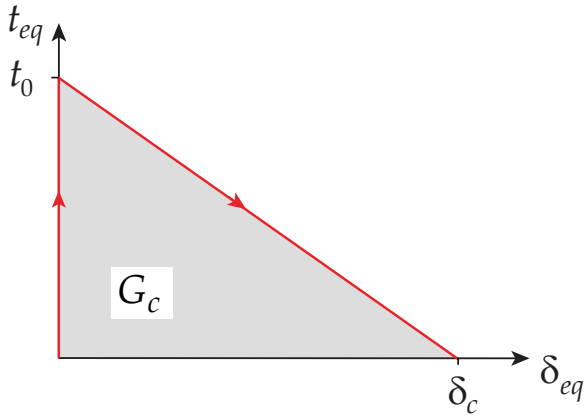


FIGURE 13.3 – Loi cohésive

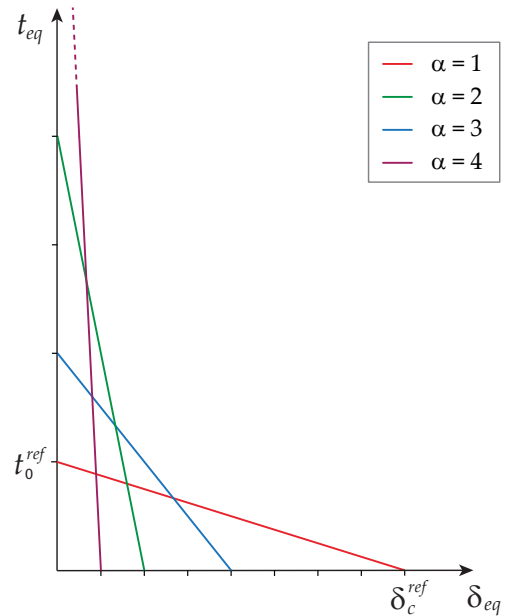


FIGURE 13.4 – Convergence d'une loi cohésive vers un modèle de Griffith

Passerelle cohésif vers Griffith On considère une loi cohésive linéairement décroissante, se traduisant par la loi suivante :

$$t_{eq} = \frac{(\llbracket u \rrbracket_c - \llbracket u \rrbracket_{eq}) t_0}{\llbracket u \rrbracket_c}, \quad (13.6)$$

$\llbracket u \rrbracket_c$ et t_0 étant des paramètres du matériau. Cette loi est affichée sur la Fig. 13.4. Imaginons que $\llbracket u \rrbracket_c$ et t_0 sont calculés à partir des paramètres de référence $\llbracket u \rrbracket_c^{ref}$ et t_0^{ref} suivants :

$$\begin{aligned} \llbracket u \rrbracket_c &= \llbracket u \rrbracket_c^{ref} / \alpha, \\ t_0 &= \alpha t_0^{ref}. \end{aligned}$$

Avec cette loi, l'énergie dissipée surfacique G_c ne dépend pas de α et s'écrit :

$$G_c = \frac{1}{2} \llbracket u \rrbracket_c^{ref} t_0^{ref}.$$

Si $\alpha \rightarrow \infty$ la fissure tend à être soumise à des conditions aux limites de bord libre. Le modèle converge donc vers un modèle de Griffith d'énergie de surface G_c . Ceci est illustré sur la Fig. 13.4 pour 4 valeurs de α . On voit donc que le modèle de Griffith ne permet pas de gérer l'initiation des fissures.

Raideur initiale La raideur d'une loi cohésive précède son initiation est théoriquement infinie. Cependant, pour des raisons numériques, les lois cohésives implémentées dans *Abaqus* comportent une raideur initiale non-nulle. Cette raideur se traduit par la définition d'une matrice de raideur reliant les composantes de la traction cohésive \vec{t} aux composantes du saut de déplacement $\vec{\delta}$:

$$\begin{pmatrix} t_n \\ t_s \\ t_t \end{pmatrix} = (1-d) \begin{pmatrix} k_{nn} & k_{ns} & k_{nt} \\ k_{ns} & k_{ss} & k_{nt} \\ k_{nt} & k_{ns} & k_{tt} \end{pmatrix} \begin{pmatrix} \llbracket u \rrbracket_n \\ \llbracket u \rrbracket_s \\ \llbracket u \rrbracket_t \end{pmatrix}.$$

Dans cette équation, d représente une variable d'endommagement, avec $d = 0$ avant l'initiation de la fissure et $d = 1$ lorsque la zone cohésive est complètement rompue.

Propagation des fissures Il existe différents critères de propagation des fissures implémentés dans *Abaqus* pour les modèles cohésifs. Pour ces TP, on utilisera principalement le *Maximum Principal Stress Criterion (MAXPS)*. Avec ce critère la direction \vec{n} du repère global associé à la fissure est alignée avec la direction de la contrainte principale la plus élevée.

$$\text{Arrêt de la fissure : } \langle t_n \rangle < \sigma_c, \quad (13.8a)$$

$$\text{Propagation de la fissure : } \langle t_n \rangle = \sigma_c, \quad (13.8b)$$

l'opérateur $\langle \cdot \rangle$ étant défini par :

$$\begin{aligned} \langle x \rangle &= x, & \text{si } x > 0, \\ \langle x \rangle &= 0, & \text{sinon,} \end{aligned}$$

$\forall x \in \mathbb{R}$. Les autres critères de propagation de zones cohésives implémentés dans *Abaqus* sont documentées dans l'*Analysis User's Guide* [2], Section 10.7.1 \rightarrow *Crack nucleation and direction of crack extension*.

2 Modèles continus

Fissuration diffuse/endommagement La dégradation du matériau se fait de façon continue dans une bande, en fonction d'une variable d'endommagement D telle que :

$$\sigma = (1 - D) C \varepsilon.$$

Cette équation peut être ré-écrite sous la forme suivante :

$$\begin{aligned} \sigma &= C (\varepsilon - \varepsilon^i), \\ \varepsilon^i &= C \varepsilon, \end{aligned}$$

ε^i représentant une déformation élastique. Une loi doit être définie pour D . Pour cela, on peut se baser sur des grandeurs équivalentes ε_{eq} et σ_{eq} définies par :

$$\begin{aligned} \varepsilon_{eq} &= \frac{1}{E} \varepsilon : C : \varepsilon, \\ \sigma_{eq} &= (1 - D) E \varepsilon : C : \varepsilon, \end{aligned}$$

Une loi de comportement $D(\varepsilon_{eq})$ peut alors être définie pour un matériau donné. À partir de cette loi, on peut représenter le comportement contrainte-déformation équivalente, une loi étant donnée en exemple sur la Fig. 13.5.

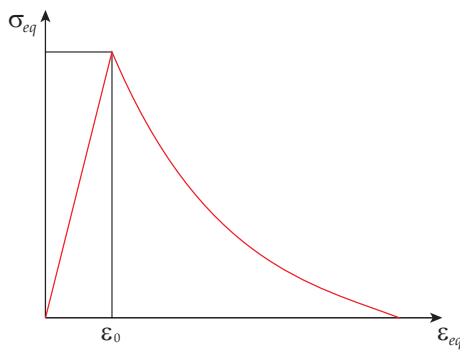


FIGURE 13.5 – Loi de comportement du matériau

Passerelle cohésif vers fissuration diffuse Un lien peut être établi avec la mécanique de la fissuration en posant :

$$\varepsilon^i = \frac{\llbracket u \rrbracket \otimes \vec{n}}{h}, \tag{13.12}$$

h étant la largeur de la zone endommagée. Cette définition de ε^i revient à considérer que la fissure est étalée dans une bande endommagée. L'équation (13.12) peut être utilisée pour convertir le saut de déplacement dans la fissure en une déformation élastique dans la bande endommagée, comme cela est illustré sur la Fig. 13.6.

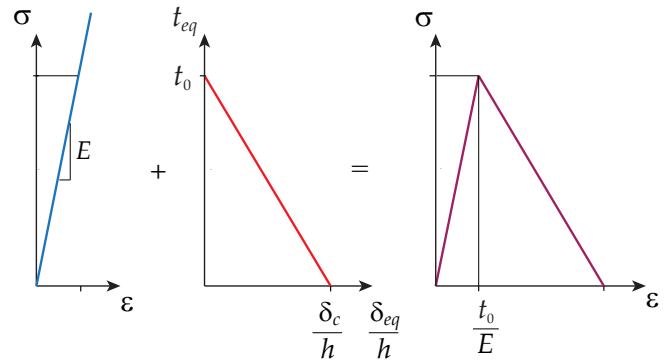


FIGURE 13.6 – Construction d'une loi de fissuration diffuse à partir d'une loi cohésive

3 Discrétisation spatiale

On a vu qu'il existe différents modèles de fissuration. Maintenant on va voir qu'il existe différentes manières de discrétiser spatialement ces modèles pour les introduire dans un calcul éléments- finis.

3.1 Fissure inter-éléments : *Seam Crack* et *Contour Integrals*

La méthode *Seam Crack* permet de dédoubler les nœuds du maillage le long d'une courbe longeant le bord de certains éléments-finis. Un nœud dédoublé reste attaché aux éléments d'un bord de la fissure et l'autre nœud dédoublé reste attaché aux éléments de l'autre bord de la fissure. Il est à noter que cette méthode telle qu'elle est implémentée dans Abaqus ne permet pas aux fissures de se propager. Il est cependant possible de calculer le facteur d'intensité des contraintes en pointe de fissure.

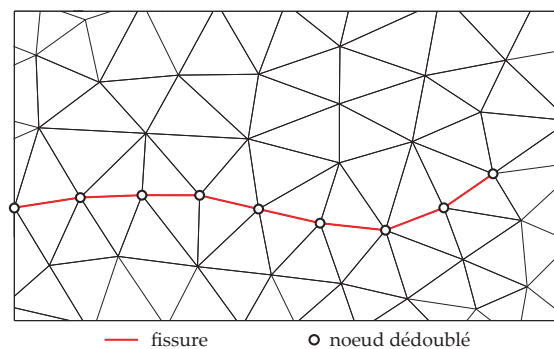


FIGURE 13.7 – Maillage avec dédoublement des nœuds sur le chemin de la fissure

3.2 Fissures d'Interface

Un problème de fissuration discontinu est semblable à un problème de contact et l'implémentation numérique d'un problème de contact peut assez facilement être étendue à

la simulation de problème de rupture cohésive. Cette approche est disponible dans *Abaqus*. Elle nécessite que le trajet de fissuration soit connu par avance et que le maillage est construit de telle sorte que la fissure passe entre les éléments du maillage. Les fissures d'*Interface* sont compatibles avec la méthode *VCCT* et un *Cohesive Behavior*. Avec la méthode *VCCT* il faut spécifier une zone de *Bonding* moins étendue que la surface totale du contact, afin de placer une pré-fissure dans le domaine nécessaire à ce que la fissure se propage en *Mécanique Élastique Linéaire de la Rupture*. Avec un *Cohesive Behavior*, il n'est pas nécessaire de prévoir de pré-fissure. Par contre, il peut être nécessaire de mailler plus finement le maillage pour obtenir la convergence à chaque pas de temps. Il faut également prévoir une légère ouverture de la zone de contact dans les zones non encore initiées.

3.3 Fissure intra-elements : X-Fem

Avec la méthode des éléments-finis étendus (eXtended Finite Element Method, X-FEM), les éléments du maillage sont coupés par la fissure. Du point de vue de l'intégration numérique, un élément coupé par la fissure se subdivise alors en deux éléments, un sous-élément appartenant à chaque bord de la fissure. Pour chacun des sous-éléments créés, les intégrations numériques se font en intégrant uniquement sur la partie de l'élément appartenant à un coté de la fissure, ceci nécessite définir de nouvelles règles d'intégration pour les sous-éléments. Du point de vue de l'approximation du champ de déplacement, la fissure est introduite via l'ajout de fonctions de forme additionnelles comportant un discontinuité au niveau de la fissure.

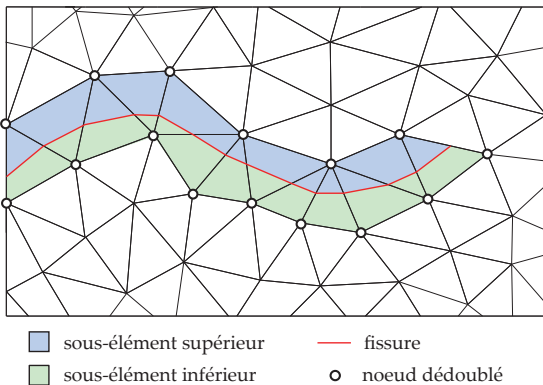


FIGURE 13.8 – Maillage coupé par une fissure

- noeud original
- noeud fantôme

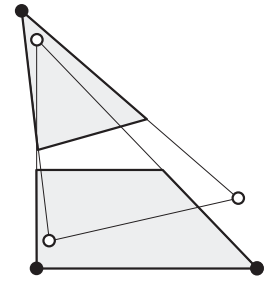


FIGURE 13.9 – Deux sous-éléments de référence de part et d'autre de la fissure

4 Discrétisation temporelle

La simulation d'un problème de fissuration peut nécessiter des adaptations particulières au niveau de la discrétisation en temps. Pour illustrer ce problème, on considère une structure sollicitée en un point par un déplacement imposé U dont la courbe force-déplacement est représentée sur la Fig. 13.10. Le premier cas de figure est celui d'une propagation stable de la fissure. On peut reconnaître ce cas de figure en regardant la courbe force-déplacement si la vitesse du déplacement reste positive. On peut alors utiliser dans *Abaqus* un *Step* de type *Static*. Dans le cas contraire, il faut utiliser une méthode de pilotage du calcul (par exemple la méthode de Riks) ou effectuer un calcul dynamique.

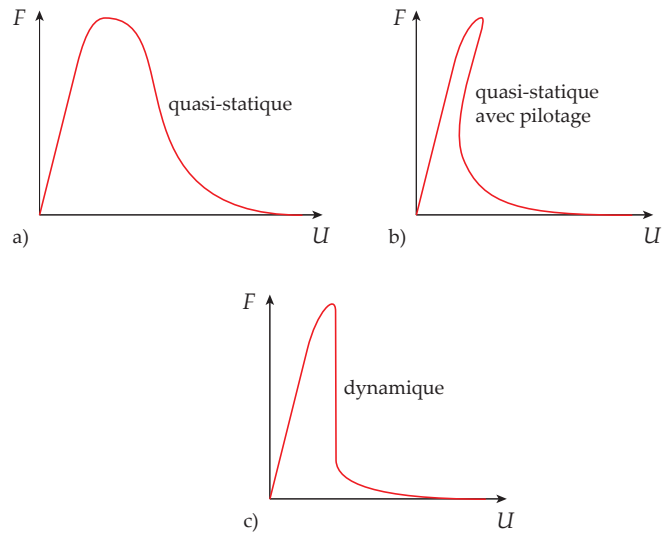


FIGURE 13.10 – Courbes force-déplacement obtenues pour différentes discrétisations temporelles

5 Documentation *Abaqus*

La documentation d'*Abaqus 6.14* nommé *USER'S GUIDE* [1] est disponible sur internet peut être consultée directe-

ment sur un navigateur en suivant ce [lien](#). Dans le *PART IV - Modeling techniques*. On peut notamment consulter les chapitres suivants :

1. Part IV : Modeling techniques, Chap. 21 : *Adhesive joints and bonded interfaces*.
2. Part IV : Modeling techniques, Chap. 31 : *Fracture Mechanics*.

On peut également se référer au *ANALYSIS USER'S GUIDE* en 5 volumes [2] également consultable sur internet ([lien](#)). On peut notamment consulter les chapitres suivants :

1. Vol. II, Part IV : Analysis techniques,
 - (a) Chap. 10 : Modeling Abstractions, Modeling discontinuities as an enriched feature using the extended finite element method (10.7.1).
 - (b) Chap. 11 : Special-Purpose Techniques, Fracture mechanics (11.4.1-11.4.3).

2. Vol. III, Part V : Materials,
 - (a) Chap. 23 : Inelastic Mechanical Properties, Concrete (23.6.1-23.6.3).
 - (b) Chap. 24 : Progressive Damage and Failure, Progressive Damage and Failure : overview (24.1.1).
 - (c) Chap. 24 : Progressive Damage and Failure, Damage and failure for ductile metals.
3. Vol. IV, Part VI : Elements,
 - (a) Chap. 32 : Special-Purpose Elements, Cohesive elements (32.5.1-32.5.10).
4. Vol. V, Part IX : Interactions,
 - (a) Chap. 37 : Contact Property Models, Surface-based cohesive behavior (37.1.10).

TP1 : Modèle de Griffith et champs asymptotiques

Introduction

Pour ce premier TP, l'objectif est de simuler un problème de fissuration pour lequel la solution analytique est connue. Vérifier la qualité des champs obtenus avec les méthodes *Seam Crack* et la *X-FEM*. Mesurer l'intégrale J .

1 Champs asymptotiques

Problème théorique On s'intéresse à une fissure de longueur $2a$ située dans un domaine élastique s'étendant jusqu'à l'infini (voir Fig. 14.1). Loin de la fissure, la structure est soumise à une traction verticale t_∞ . On se place dans le cadre de l'hypothèse des contraintes planes. La solution analytique de ce problème, appelée solution de Westergaard, est connue.

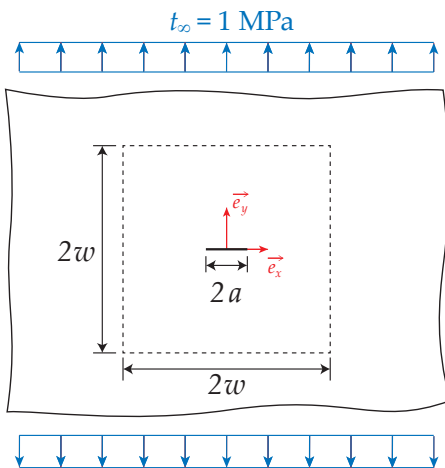


FIGURE 14.1 – Problème de Westergaard : une fissure de longueur a dans un domaine s'étendant à l'infini

La solution exacte du déplacement u_y s'écrit :

$$u_y = t_\infty \frac{1 + \nu}{2E} \sqrt{r_1 r_2} \sin\left(\frac{\theta_1 + \theta_2}{2}\right) (\kappa + 1).$$

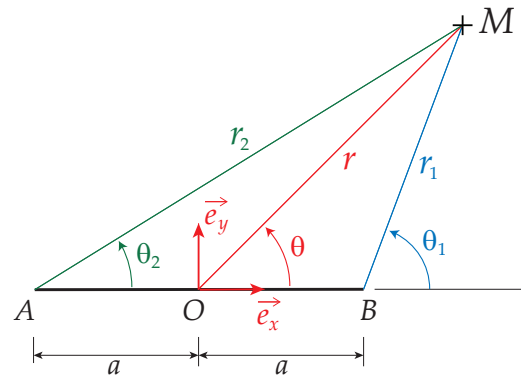


FIGURE 14.2 – Paramétrage pour l'écriture la solution asymptotique au voisinage de la pointe

avec :

$$\begin{aligned} \kappa &= 3 - 4\nu, && \text{en déformations planes,} \\ \kappa &= (3 - \nu)/(1 + \nu), && \text{en contraintes planes,} \end{aligned}$$

(r_1, θ_1) et (r_2, θ_2) les coordonnées polaires autour des extrémités points A et B de la fissure, respectivement, telles que définies sur la Fig. 14.2. Sur la surface de la fissure, on obtient :

$$u_y = t_\infty \frac{1 + \nu}{2E} \sqrt{a^2 - x^2} (\kappa + 1).$$

Cette expression peut se retrouver dans [7]. Le profil d'ouverture elliptique de la fissure correspondant est représenté sur la Fig. 14.3 (courbe rouge). Au voisinage de la pointe B de la fissure, la composante u_y du champ de déplacement s'écrit :

$$u_y = K_I \frac{(1 + \nu)}{E} \sqrt{\frac{r_1}{2\pi}} \sin\left(\frac{\theta_1}{2}\right) (\kappa + 1 - 2 \cos^2\left(\frac{\theta_1}{2}\right)),$$

avec K_I le facteur d'intensité des contraintes en mode I satisfaisant :

$$K_I = t_\infty \sqrt{\pi a}.$$

Sur le bord Γ_c^+ , la composante verticale du déplacement s'écrit :

$$u_y = K_I \frac{1 + \nu}{E} \sqrt{\frac{(a - x)}{2\pi}} (\kappa + 1).$$

On obtient donc une ouverture de forme parabolique. Le taux de restitution d'énergie élastique G s'obtient ensuite en écrivant dans le cas de contraintes planes :

$$G = \frac{K_I^2}{E}$$

Problème défini sur un carré On souhaite comparer ces résultats analytiques avec des résultats numériques obtenus dans *Abaqus*. Il n'est pas possible de mailler un domaine s'étendant jusqu'à l'infini, on définit donc à la place un domaine d'étude carré de côté $2w$. Il est possible de corriger le facteur d'intensité des contraintes pour tenir compte de la géométrie du problème. Un facteur correctif α peut être défini de la manière suivante :

$$\alpha = \cos\left(\frac{\pi a}{2w}\right)^{-1/2},$$

À l'aide de ce facteur correctif, on peut calculer une traction corrigée $(t_\infty)_{cor}$ et un facteur d'intensité des contraintes corrigé $(K_I)_{cor}$ de la manière suivante :

$$(t_\infty)_{cor} = \alpha t_\infty,$$

$$(K_I)_{cor} = \alpha K_I,$$

w étant la largeur du domaine d'étude. On veut étudier la déformation de cette structure en utilisant deux méthodes : la méthode *SEAM crack* et la méthode *X-FEM*.

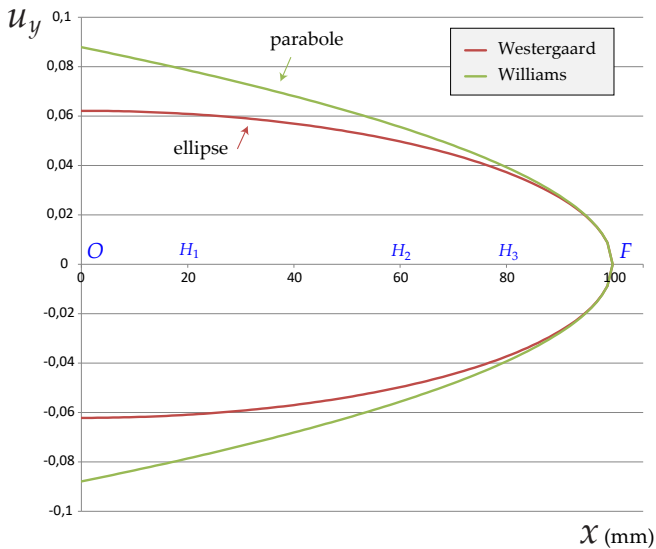


FIGURE 14.3 – Solution analytique de l'ouverture de la fissure obtenus en prenant en compte le facteur correctif α

2 Données du problème

Géométrie On considère la géométrie donnée sur la Fig. 14.4a. Les dimensions de la fissure et du domaine sont

la longueur de la fissure noté a , la largeur qui est aussi la demi-hauteur notée w , et l'épaisseur notée e , avec :

$$a = 100 \text{ mm},$$

$$w = 500 \text{ mm},$$

$$e = 1 \text{ mm}.$$

En exploitant la symétrie du problème par rapport à un axe vertical, seulement un demi-problème est discrétisé.

Matériau Le matériau considéré est du béton. Les paramètres mécaniques sont le module d'Young E , le coefficient de Poisson ν , et l'énergie surfacique de rupture G_c dont les valeurs sont :

$$E = 3300 \text{ MPa},$$

$$\nu = 0.37,$$

$$G_c = 0.5 \text{ N/mm}.$$

La valeur de G_c ne sera pas utilisée dans le premier TP car on ne s'intéresse pas encore à la propagation des fissures, qui sera étudiée dans les TP suivants.

Conditions aux limites Le problème est supposé plan avec l'hypothèse des contraintes planes. Les conditions aux limites sont données sur la Fig. 14.4b avec :

$$t_\infty = 1 \text{ MPa}.$$

3 Utilisation du tableur

Questions Dans un tableur :

1. Calculer la valeur du facteur d'intensité des contraintes K_I obtenue pour une plaque infinie et la valeur corrigée $(K_I)_{cor}$ pour une plaque de dimension finie.

Les valeurs de K_I pour une dalle infinie et la valeur corrigée pour un domaine carré sont données dans le Tableau 14.1. La valeur de α obtenue est la suivante :

$$\alpha = 1.02540832.$$

	MPa. $\sqrt{\text{mm}}$
K_I	17.72453...
K_I corrigé	18.17488...

TABLE 14.1 – Facteurs d'Intensité des Contraintes (FIC)

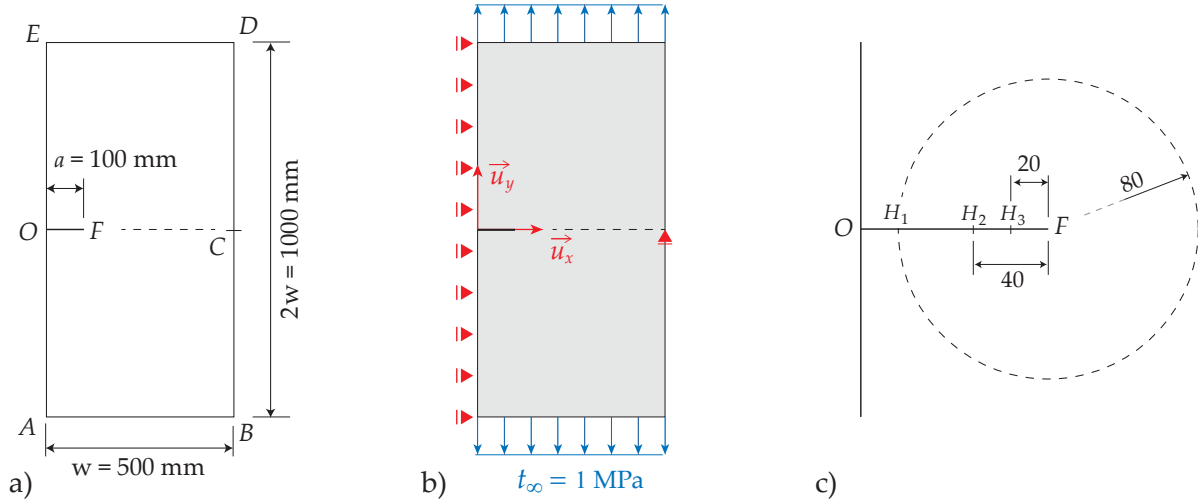


FIGURE 14.4 – a) Géométrie, b) Conditions aux limites, c) données pour le *Sketch*

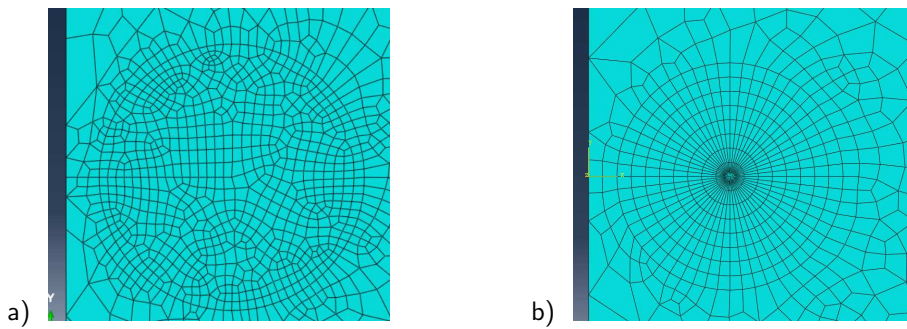


FIGURE 14.5 – a) maillage du disque avec l'option *Free*, b) maillage réalisé avec l'option *Sweep*

2. Calculer les valeurs correspondantes G et G_{cor} du taux de restitution d'énergie élastique.

On obtient les valeurs de G données dans le Tableau 14.2.

	N/mm
G	0.0952
G corrigé	0.1001

TABLE 14.2 – Taux de restitution d'énergie élastique

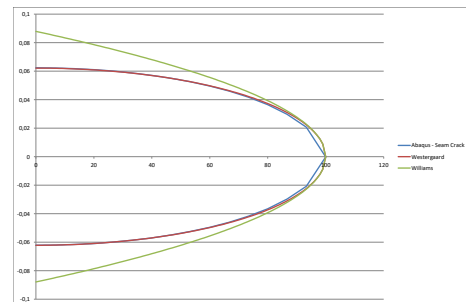


FIGURE 14.6 – Solution analytique de l'ouverture de la fissure

3. Tracer le champ $u_y(x)$ pour $x \in [0, a]$ donné par la solution de Westergaard. On pourra se baser sur un découpage de l'intervalle $[0, a]$ en 100 segments d'égaux longueurs.

4. Tracer le champ $u_y(x)$ pour $x \in [0, a]$ donné par la solution de Williams (solution asymptotique).

4 Calculs dans Abaqus

4.1 Test 1 : méthode Seam Crack

Réaliser un calcul en utilisant la méthode *Seam crack* pour laquelle la fissure passe entre les éléments du maillage.

Questions

- Renommer le modèle *Model-1-Seam-Crack*
- Dans le module *Part* :
 - Créer un *Part* pour le contour du domaine étudié reliant les points O , A , B , C , D , et E . Dans le *Sketch*, ne pas oublier de définir le point milieu C du segment $[BD]$ pour pouvoir plus tard imposer un déplacement vertical nul en ce point.
 - Dans le menu *Tools* → *Partition*, choisir *Type* → *Face* et *Method* → *Sketch* pour créer la surface de la fissure et le cercle de centre F ainsi que les points H_1 , H_2 , et H_3 . Ceci imposera lors de la création du maillage que le bord des éléments suive le trajet de la fissure. Cependant, une opération supplémentaire sera nécessaire pour dédoubler les nœuds de la fissure.
 - Créer un *Set* nommé *Set-Crack* pour la surface fissurée.
 - Créer trois *Sets* nommés *Set-H1*, *Set-H2*, et *Set-H3* pour les points H_1 , H_2 , et H_3 , respectivement.
- Dans le module *Property* :
 - Définir un *Material* de type *Elastic* → *Isotropic*.
 - Définir une *Section* de type *Homogeneous* puis retourner dans le *Part* faire le *Section assignment*.
- Dans le module *Assembly* :
 - Créer un assemblage avec le *Part* qui a été créé. Choisir l'option *Independent* pour que le maillage soit indépendant du *Part*. Ceci est nécessaire pour créer un *Seam crack*.
- Dans le module *Mesh* :
 - En utilisant l'outil *Seed* → *Part* :
 - Spécifier une densité globale de 80 mm à l'extérieur du cercle de centre F .
 - En utilisant l'outil *Seed* → *Edges* :
 - Spécifier une densité de 7 mm sur le cercle de centre F et le long des segments $[H_1H_2]$, $[H_2H_3]$, et $[H_3F]$.
 - Spécifier une densité de 7 mm le long du segment $[OH_1]$,
 - Utiliser l'outil *Assign Mesh Controls* pour spécifier sur tout le domaine les options *Element Shape* → *Quad*, *Technique* → *Free* et *Algorithm* → *Advancing front*.
 - Utiliser l'outil *Element Type* pour spécifier des éléments quadrangles linéaires en décochant l'option *Reduced Integration*.
 - Utiliser l'outil *Mesh Part* pour créer le maillage.
- Dans le module *Interaction* :
 - Dans le menu déroulant *Special* → *Crack*, choisir *Assign Seam*. Sélectionner le *Set* qui a été créé pour la fissure. Les nœuds de ce *Set* sont alors dédoublés.
 - Pour vérifier que le *Seam* a bien été créé, dans le menu déroulant supérieur *Tools* → *Query*. Créer un rectangle de sélection autour d'un nœud sur la fissure. Cliquer sur *Done* et constater dans la console que deux nœuds ayant les mêmes coordonnées ont bien été sélectionnés.
 - Définir un calcul d'intégrale J à partir du menu déroulant *Special* → *Crack* → *Create* → *Contour integral*. Lorsque *Abaqus* demande de définir un *Set* pour le *Crack Front*, indiquer le disque de centre F . Lorsque *Abaqus* demande le *Crack Tip Region*, indiquer F . Indiquer ensuite la direction *Normal to the crack* en définissant par ses coordonnées le vecteur $-\vec{e}_y$.
- Dans le module *Step* :
 - Créer un *Step* de type *Static, General* en vue de réaliser un calcul élastique.
 - Utiliser l'outil *Create History Output* → *Create* pour créer un *History Output* nommé *H-Output-J* en choisissant *Domain* → *Crack*. Spécifier *Number of contours* = 1. Ceci permettra de calculer numériquement l'intégrale J lors du calcul.
 - Créer un autre *History Output* nommé *H-Output-K_I* pour le calcul du facteur d'intensité des contraintes K_I .
- Dans le module *Load* :
 - Imposer les conditions aux limites en déplacement le long des segments $[AO]$ et $[OE]$.
 - Imposer les conditions aux limites en déplacement au point C .
 - Imposer les conditions aux limites de traction sur le bord inférieur et sur le bord supérieur.
- Dans le module *Job* :
 - Créer un *Job* nommé *Job-1-Seam-Crack*.
 - Lancer un *Job*.
- Dans le module *Visualization* :
 - Depuis l'arborescence de gauche, créer un *Path* de type *Node list* pour la fissure. Dans la fenêtre *Edit Node List Path*, cliquer sur *Node set selections* → *Add After...* puis sélectionner *PART-1-1.SET-CRACK*.
 - Utiliser l'outil *Create XYData* → *Path* pour extraire le champ de déplacement vertical u_y le long de la fissure. Le tracer la courbe correspondant dans le tableur en la superposant avec la solution de Westergaard et la solution de Williams.

- (c) Utiliser l'outil *Create XYData* → *ODB history output* pour lire la valeur l'intégrale J et du facteur d'intensité des contraintes K_I . Comparer avec les valeurs théoriques calculées dans le tableur. Calculer l'erreur en pourcentage par rapport à la valeur théorique corrigée.

4.2 Test 2 : Maillage radial

Dans un nouveau *Model*, nous allons redéfinir les densités de telle sorte que le maillage soit plus fin au voisinage de la pointe de la fissure.

Questions

- Dupliquer le *Model-1-Seam-Crack*. Renommer le nouveau modèle *Model-2-Radial*.
- Dans le module *Mesh* :
 - Utiliser l'outil *Assign Mesh Controls* les options sur le disque entourant la pointe F de la fissure pour sélectionner les options *Element Shape* → *Quad-dominated* et *Technique* → *Sweep*.
 - Utiliser l'outil *Seed Edges* pour spécifier une densité de 9 mm le long des segments $[OH_1]$, $[H_1H_2]$, $[H_1H_2]$, $[H_1H_2]$, et le long du cercle centré sur F .
 - Remailler le disque entourant la pointe F de la fissure. Ceci permet d'obtenir un maillage radial tel que représenté sur la Fig. 14.5b.
- Dans le module *Interaction* :
 - Dupliquer les nœuds le long du trajet de la fissure.
- Dans le module *Job* :
 - Créer un *Job* nommé *Job-2-Radial*.
 - Lancer le calcul. Le message d'erreur suivant devrait apparaître :


```
Triangles, Tetrahedra and wedges may not be used in a contour integral region. 56 elements are within a contour region. The elements are identified in element set ErrElemIncorrContInteg.
```
 - Supprimer une fonctionnalité du calcul pour faire en sorte que le calcul fonctionne.
- Dans le module *Vizualization* :
 - Extraire à nouveau le champ u_y sur le bord de la fissure, le tracer dans le tableur en le superposant aux autres courbes théoriques et numériques.

4.3 Test 3 : Méthode X-FEM

Réaliser un calcul en utilisant cette fois la méthode *X-FEM* au lieu de la méthode *Seam Crack* pour définir la fissure.

Questions

- Créer un nouveau *Model* nommé *Model-3-XFEM*.
- Dans le module *Part* :
 - Créer un *Part* de type *Shell* pour la structure. Placer le point C dans le *Sketch* pour appliquer les conditions aux limites par la suite, mais ne pas placer le point O sur le contour du domaine car avec la méthode *X-FEM* il vaut mieux qu'aucun nœud se trouve sur le trajet de la fissure.
 - Créer un *Part* de type *Deformable* et *Wire* contenant la géométrie de la fissure (ligne horizontale de longueur 100 mm).
- Dans le module *Properties* :
 - Définir un *Material* de type *Elastic* → *Isotropic*.
 - Ajouter dans le *Material* une partie *Damage For Traction Separation Laws* → *MAXPS Damage* (même si il n'y a pas de propagation de fissure pour ce test). Choisir un paramètre très grand pour *Max Principal Stress* (par exemple 10000000) et dans *Suboptions* → *Damage Evolution* donner une valeur quelconque pour *Displacement at failure* (par exemple 1000).
- Dans le module *Mesh* :
 - Utiliser l'outil *Seed Edges* pour imposer les nombres d'éléments suivants :
 - 41 le long du segment $[AE]$.
 - 20 le long des segments $[ED]$, $[DC]$, et $[AB]$.
 - 21 le long du segment $[CB]$.
 - Utiliser l'outil *Mesh Controls* pour spécifier les options de maillage *Element Shape* → *Quad*, *Technique* → *Free*, et *Algorithm* → *Advancing front*.
 - Utiliser l'outil *Assign Element Type* pour spécifier des éléments de type quadrangles en décochant l'option *Reduced Integration*.
 - Utiliser l'outil *Mesh Part* pour créer le maillage.
- Dans le module *Assembly* :
 - Créer un assemblage avec les deux *Parts*.
 - Définir le positionnement relatif des deux *Parts* avec la fonction *Translate Instance* du module *Assembly*.
- Dans le module *Interaction* :
 - Dans le menu déroulant supérieur sélectionner *Crack* → *Create*. Dans la fenêtre *Create Crack* qui s'ouvre, créer une fissure de type *XFEM*. Pour le *Crack Domain*, spécifier la structure entière de l'éprouvette rectangulaire. Cocher *Crack location* et sélectionner le *Part* de la fissure. Ne pas cocher *Specify contact property* car la loi de comportement de la fissure a déjà été définie dans le *Material*.
- Dans le module *Step* :
 - Créer un *Step* pour réaliser un calcul élastique.
 - Aller dans *Field Output Request* et dans *Failure/Fracture*, cocher les cases *PHILSM* et *PSILSM*. Ceci est important pour que la fissure soit bien visible par la suite dans le module *Visualization*.

8. Dans le module *Job* :

- (a) Créer un *Job* nommé *Job-3-XFEM*.
- (b) Lancer un *Job*.

(a) Est-il possible de mesurer l'ouverture de la fissure ?

(b) Que dire des conditions aux limites appliquées sur les éléments rompus au voisinage du bord gauche ?

9. Dans le module *Visualization* :

TP2 : Fissuration entre les éléments du maillage

Introduction

Dans ce TP, on étudie à nouveau l'éprouvette trapézoïdale vue dans le TP *Interaction-Contact*. Cette fois, l'objectif est d'étudier la fissuration dans l'éprouvette lorsque celle-ci est chargée. Des calculs seront effectués en supposant le trajet de la fissure vertical et en faisant passer la fissure entre les éléments du maillage. Les résultats des calculs seront comparés à des résultats expérimentaux [3].

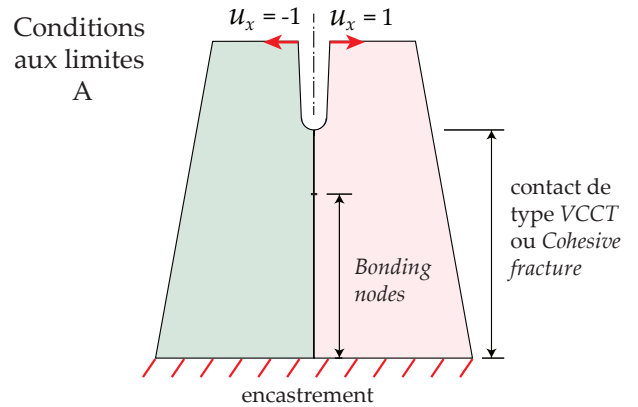


FIGURE 15.2 – Conditions aux limites **A** appliquées sur l'éprouvette

1 Données du problème

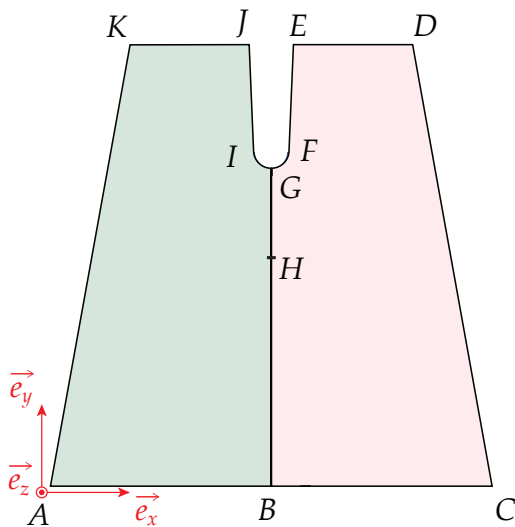


FIGURE 15.1 – Définitions des points A, B, \dots, K le long du contour de l'éprouvette

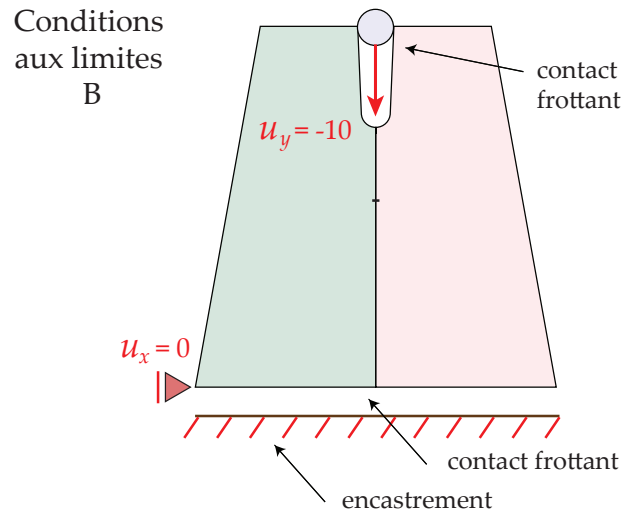


FIGURE 15.3 – Conditions aux limites **B** appliquées sur l'éprouvette.

Géométrie La géométrie des 4 Parts du problème étudié peut être obtenue dans l'annexe *Code JNL pour le TP2*, ce code pouvant être copié et collé dans la console d'*Abaqus*. L'épaisseur de l'éprouvette trapézoïdale est de 10 mm.

Matériau Le matériau constituant l'éprouvette est du *Plexiglas* [4]. Les valeurs numériques du module d'Young E , du coefficient de Poisson ν , et de l'énergie l'énergie sur-

faciende de rupture G_c sont les suivantes :

$$\begin{aligned} E &= 3300 \text{ MPa}, \\ \nu &= 0.37, \\ G_c &= 0.5 \text{ N.mm}^{-1}. \end{aligned}$$

Plus précisément, les paramètres de définition de la VCCT avec un *Power law* seront :

$$\begin{aligned} G_c^m &= G_c^s = G_c^t = 0.5 \text{ N.mm}^{-1}, \\ \alpha &= 1. \end{aligned}$$

La *traction à la rupture* est théoriquement $t_0 = 72 \text{ MPa}$, mais dans la pratique pour éviter les problèmes de convergence lorsque le maillage est grossier, on choisira :

$$t_0 = 10 \text{ MPa}.$$

La loi cohésive est supposée linéairement décroissante. Par ailleurs, le coefficient de frottement du PMMA sur l'acier est :

$$f = 0.2.$$

Conditions aux limites On se place dans le cadre de l'hypothèse des contraintes planes. Deux types de conditions aux limites présentées ci-dessous seront considérées pour ce TP.

— *Conditions aux limites A* : Au niveau des points J et E , on impose :

$$\begin{aligned} u_x(J) &= -1 \text{ mm}, \\ u_x(E) &= 1 \text{ mm}. \end{aligned}$$

Au niveau des segments $[AB]$ et $[BC]$, on impose un encastrement, donc :

$$\begin{aligned} u_x &= 0 \text{ mm}, \\ u_y &= 0 \text{ mm}. \end{aligned}$$

— *Conditions aux limites B* : Au niveau du *Reference Point* du *Punch*, on impose :

$$\begin{aligned} u_x &= 0 \text{ mm}, \\ u_y &= -10 \text{ mm}, \\ \theta &= 0. \end{aligned}$$

Au niveau du *Reference Point* du *Ground*, on impose un encastrement :

$$\begin{aligned} u_x &= 0 \text{ mm}, \\ u_y &= 0 \text{ mm}, \\ \theta &= 0. \end{aligned}$$

Par ailleurs, on spécifie un contact avec un coefficient de frottement $f = 0.45$ au niveau des contacts *Ground/ HalfTrapezoid-Left*, *Ground/ HalfTrapezoid-Right*, *Punch/ HalfTrapezoid-Left*, et *Punch/ HalfTrapezoid-Right*.

2 Calculs dans *Abaqus*

Test 1 : Fissure de type VCCT (version 6.14)

À la différence de l'éprouvette déjà étudiée pour le problème de contact, l'éprouvette étudiée dans ce TP comporte une pré-fissure nommée *CrackInit* de longueur 10 mm déjà rompue avant que l'essai ait commencé. La présence du *CrackInit* permet expérimentalement d'éviter le cas d'une propagation instable de la fissure. Numériquement, sa présence est nécessaire avec la méthode VCCT qui ne permet pas de traiter l'initiation des fissures. On commence par simuler un test (*Test 1A*) avec le modèle VCCT en utilisant des conditions aux limites **A** représentées sur la Figure 16.1a. L'éprouvette est décomposée dès le départ en deux *Parts* nommés *HalfTrapezoid-Right* et *HalfTrapezoid-Left*, comme cela est indiqué sur la Fig. 16.4a, la fissure se propageant entre ces deux *Parts* dans le prolongement du *CrackInit*.

Questions

1. Importer la géométrie de l'éprouvette depuis le fichier TP_demiTrap.txt en copier et collant le contenu du fichier dans la console. Un modèle nommé *InterElemFrac* contenant 4 *Parts* devrait apparaître dans l'arborescence de gauche.
2. Renommer ce modèle *CondLimA-VCCT* et supprimer le *Model-1* qui ne doit rien contenir.
3. Dans le module *Part* :
 - (a) Observer la différence entre les *Surfaces Surf-CrackTot-Left*, *Surf-CrackCalc-Left*, et *Surf-CrackIni-Left*. Voir aussi les *Sets* définis sur les bords droits et gauches de la fissure. L'ensemble des surfaces nommées est représenté sur la Fig. 16.4.

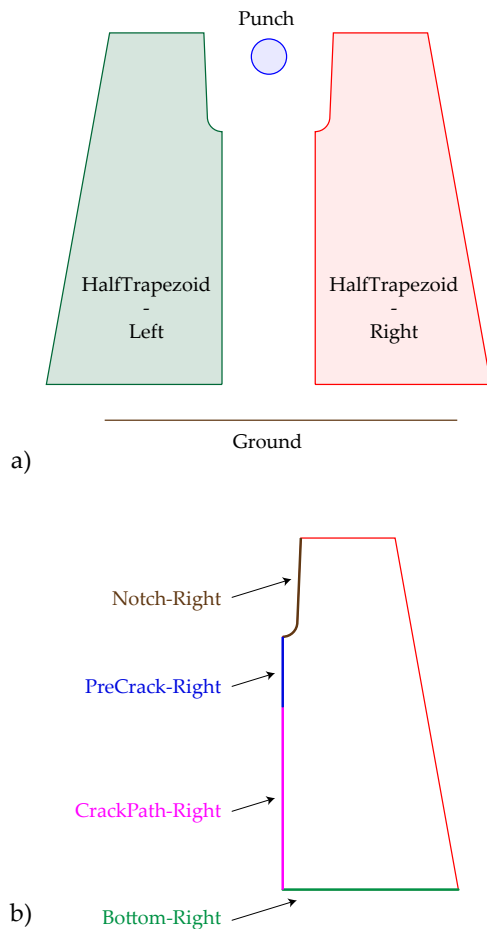


FIGURE 15.4 – a) Les 4 *Parts* définis dans *Abaqus* et b) les noms des *Sets* et des *Surfaces*

4. Dans le module *Property* :

- (a) Utiliser l'outil *Create Material* pour définir le *Material* de type *Mechanical* → *Elasticity* → *Elastic* composant les 2 demi-trapèzes.
- (b) Utiliser l'outil *Create Section* pour créer la *Section*. Ne pas oublier de spécifier l'épaisseur de l'éprouvette.
- (c) Utiliser l'outil *Assign Section* pour assigner la *Section* aux deux demi-trapèzes.

5. Dans le module *Mesh* :

- (a) Sélectionner le *Part HalfTrapezoid-Left*
- (b) Utiliser l'outil *Assign Mesh Controls* pour spécifier un maillage triangulaire.
- (c) Définir une densité globale de 2.5 mm.
- (d) Au niveau des courbes $[GI]$, $[IJ]$, $[GH]$, et $[HB]$, spécifier une densité plus fine de 1 mm.
- (e) Répéter l'opération avec le *Part HalfTrapezoid-Right*.

6. Dans le module *Assembly* :

- (a) Créer dans l'assemblage une *Instance* du *Part HalfTrapezoid-Left* et une *Instance* du *Part HalfTrapezoid-Right*.

7. Dans le module *Step* :

- (a) Créer un *Step* de type *Static, General*.
- (b) Dans l'onglet *Incrementation*, choisir *Type* : *Automatic* et *Maximum number of increments* : 100.

8. Dans le module *Load* :

- (a) Imposer la condition aux limites d'encastrement sur les segments $[AB]$ et $[BC]$.
- (b) Imposer le déplacement suivant \vec{e}_x au point J .
- (c) Imposer le déplacement suivant \vec{e}_x au point E .

9. Dans le module *Interaction* :

- (a) Utiliser l'outil *Create Interaction Property* pour créer une *Interaction Properties* : Dans l'encart *Type*, choisir *Contact* et cliquer sur *Continue...* Créer une *Contact Property Option* de type *Mechanical* → *Fracture Criterion*. Dans l'encart *Fracture Criterion*, choisir les paramètres suivants :
 - Type : *VCCT*,
 - Mixed mode behavior : *BK*,
 - Critical energy release rate : mode I = 0.5 N.mm⁻¹, mode II = 0.5 N.mm⁻¹, mode III = 0.5 N.mm⁻¹.
 - Exponent n : 1.
- (b) Utiliser l'outil *Create Interaction* pour créer une *Interaction* :
 - Créer une *Interaction* de type *Surface-to-surface contact*.
 - Sélectionner le bord gauche de la fissure comme *Master Surface* et le bord droit de la fissure comme *Slave Surface*. Il est important que ces deux surfaces contiennent également la pré-fissure, car la méthode *VCCT* ne permet de simuler l'initiation des fissures. La sélection des deux surfaces est rendue difficile par le fait que les deux surfaces sont cotes à cotes dans l'*Assembly*. Pour effectuer la sélection, on pourra utiliser l'une des 3 méthodes suivantes :
 - Une première solution est d'utiliser les *Surfaces* déjà définies dans les *Parts*. On pourra sélectionner *Surf-CrackTot-Left* comme *Master Surface* et *Surf-CrackTot-Right* comme *Slave Surface*.
 - Une autre solution est de déplacer temporairement les deux *Parts* en contact dans l'*Assembly* le temps de définir l'*Interaction*, puis de les remettre en place.
 - Une troisième solution est d'utiliser l'outil *Remove Selected* de la barre d'outils supérieure pour faire disparaître temporairement l'un des deux *Parts*. On peut ensuite utiliser l'outil *Invert Display* pour intervertir le *Part* caché et le *Part* affiché. Une fois les sélections terminées, on peut cliquer que *Replace All* pour afficher à nouveau l'ensemble des pièces.

- Dans le menu déroulant *Discretization method*, sélectionner *Node to surface* car l'option *Surface to surface* n'est pas compatible avec la méthode *VCCT*.
 - Dans l'onglet *Bonding* cocher la case *Limit bonding to slave nodes in subset* et sélectionner un *Set* contenant les nœuds de la *Slave Surface* initialement attaché à la *Master Surface*, c'est à dire *Set-CrackCalc-Right*. Ce *Set* ne comprend pas la pré-fissure, il doit donc être moins étendu que la *Slave Surface* ayant été définie.
 - Dans le menu déroulant *Contact interaction property*, sélectionner l'*Interaction Property* créée lors de l'étape précédente.
- (c) Dans le menu déroulant supérieur, sélectionner *Special* → *Cracks* → *Create*. Créer un *Crack* de type *Debond using VCCT*. Dans le menu déroulant *Initiation Step*, choisir le *Step* courant. Dans le menu déroulant *Contact pair interaction*, choisir l'*Interaction* qui a été créée pour la fissure.
10. Dans le le module *Job* :
- (a) Créer un *Job*.
 - (b) Lancer un calcul.
11. Dans le module *Visualization* :
- (a) Observer que la fissure s'ouvre bien au fur et à mesure que l'éprouvette est chargée.

Test2 : Modèle cohésif

Questions

1. Dupliquer le modèle *CondLimA-VCCT*. Renommer nouveau modèle *CondLimA-CZM* (pour *cohesive zone model*).
2. Dans le module *Assembly* :
 - (a) Dans l'arborescence de gauche, suivre le chemin *Assembly* → *Engineering Features* → *Cracks* et supprimer le *Crack-1* ayant été créé pour la méthode *VCCT*.
3. Dans le module *Interaction* :
 - (a) Supprimer l'*Interaction Property IntProp-VCCT* (faire *Clic-droit* → *Delete*).
 - (b) Créer une *Interaction Property* de type *Contact* nommée *IntProp-CZM* :
 - Dans l'encart *Contact Property Options*, créer un *Mechanical* → *Cohesive Behavior* et cocher la case *Allow cohesive behavior during repeated post-failure contacts*.
 - Créer ensuite un *Mechanical* → *Damage*. Spécifier la contrainte d'initiation dans l'onglet *Initiation* (mettre la même valeur pour *Normal Only*, *Shear-1 Only*, et *Shear-2 Only*). Spécifier ensuite l'énergie surfacique de rupture dans l'onglet *Evolution* après avoir coché la case *Specify damage evolution* et *Type* → *Energy*.

- Créer enfin un *Geometric Properties* pour spécifier la *Out-of-plane surface thickness*.
- (c) Désactiver l'interaction *Int-VCCT* (dans l'arborescence de gauche, faire *Clic-droit* → *Supprimer*).
- (d) Créer une *Interaction* de type *Surface-to-surface contact* :
 - Choisir *Sliding formulation* → *Small sliding* et faire pointer l'*Interaction* vers l'*Interaction Property* qui a été créée.

Comparaison Par rapport à la méthode *VCCT*, on peut noter les différences suivantes :

1. Avec la méthode *VCCT*, il est obligatoire de cocher l'option *Limit bonding to slave nodes in subset* et de spécifier un *Set* moins étendu que la *Slave Surface* de l'interaction. Avec une surface cohésive, cette option peut également être utilisée, mais ce n'est pas obligatoire. À la place, on peut simplement retirer la pré-fissure de la lors de la sélection de la *Master Surface* et de la *Slave Surface*.
2. Avec la surface cohésive, il n'est pas nécessaire de définir un *Crack* dans le module *Assembly*.

Test3 : VCCT avec conditions aux limites réelles

On modifie le test pour prendre en compte à la fois de l'interface avec le *Poinçon* nommé *Punch* dans les simulations et de l'interface avec le *Sol* nommé *Ground* dans les simulations. On s'attachera à tracer la courbe représentant la composante verticale de la force appliquée par le *Punch* sur un demi-domaine en fonction de l'ouverture entre les points situés aux extrémités de la pré-entaille.

Questions

1. Dupliquer le modèle *CondLimA-VCCT*. Renommer nouveau modèle *CondLimB-VCCT*.
2. Dans le module *Part* :
 - (a) Depuis le menu déroulant supérieur, sélectionner *Tools* → *Reference Point*. Créer un *Reference Point* pour le *Ground* (la position n'a pas d'importance).
 - (b) Depuis le menu déroulant supérieur, sélectionner *Tools* → *Set* → *Create*. Créer un *Set* nommé *Set-RP* pour le *Reference Point*.
 - (c) Créer également un *Reference Point* et le *Set* correspondant pour le *Punch*.
3. Dans le module *Assembly* :
 - (a) Créer une instance pour le *Part Ground-1*.
 - (b) Créer une instance pour le *Part Punch-1*.
4. Dans le module *Load* :
 - (a) Supprimer les conditions aux limites appliquées précédemment sur les segments $[AB]$ et $[BC]$, ainsi que les points E et J .

- (b) Imposer un déplacement de corps rigide du *Punch* avec $U1 = 0$, $U2 = -10$, et $UR3 = 0$.
- (c) Imposer l'encastrement du *Ground*. Dans ce cas, on bloquera le déplacement suivant \vec{e}_x et \vec{e}_y ainsi que la rotation autour \vec{e}_z .
- (d) Bloquer le déplacement suivant \vec{e}_x des points *A* et *C*.

5. Dans le module *Interaction* :

- (a) Imposer un contact entre le *Punch* et l'éprouvette :
 - Créer une *Interaction Properties* nommée *IntProp-Contact* de type *Contact* nommée *IntProp-Punch*.
 - Dans la fenêtre *Contact Property Options* ajouter un contact de type *Mechanical* → *Normal Behavior* → "Hard" *Contact*.
 - Ajouter également un contact de type *Mechanical* → *Tangential Behavior*. Choisir une *Friction formulation* de type *Lagrange Multiplier* en imposant la bonne valeur du *Friction Coeff.*
 - Finalement ajouter *Geometric Properties* avec une *Out-of-plane surface thickness* correspondant à l'épaisseur de l'éprouvette.
 - Créer une *Interaction Int-Punch* de type *Surface-to-Surface Contact* pointant vers cette *Interaction Properties* en choisissant :
 - Le cercle comme *Master surface*,
 - Les deux plans inclinés du *Trapezoid* comme *Slave surface* en cochant l'option *Surface to surface*.
 - Créer une *Interaction Int-Ground* de type *Surface-to-Surface Contact* pointant vers *IntProp-Contact*.
- (b) Imposer un contact entre le *Ground* supposé immobile et l'éprouvette.

6. Dans le module *Visualization* :

- (a) Exporter dans un tableur la courbe de la force verticale appliquée par l'éprouvette sur le *Punch* en fonction de la norme du déplacement du *Punch*.

Test4 : Modèle cohésif avec conditions aux limites réelles

Pour le *Test 4* on reprend le *Test 3* en utilisant un modèle cohésif à la place du modèle *VCCT*.

Bilan

Les courbes de traction obtenues devraient ressembler à celles présentées sur la Fig. 15.5.

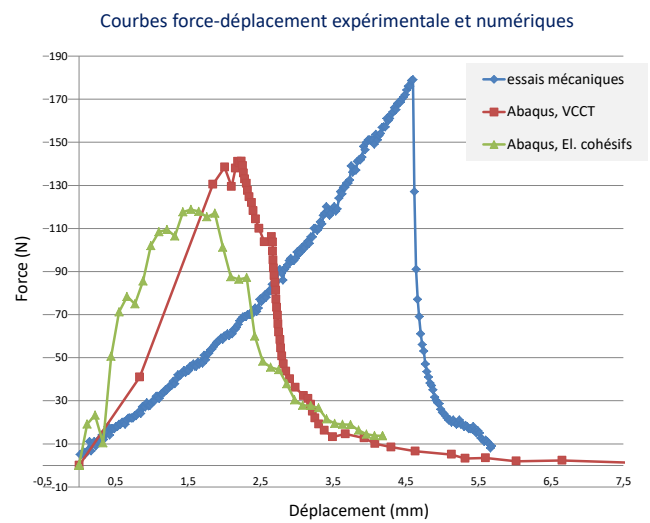


FIGURE 15.5 – Courbes force-déplacement obtenues avec les conditions aux limites **B** (courbe obtenue en prenant $f = 0.45$ et des densités de 1 mm et 0.5 mm)

TP3 : Fissuration à travers les éléments du maillage

Introduction

Dans ce TP, on reprend le test sur éprouvette trapézoïdale du TP précédent avec les mêmes conditions aux limites nommées **A** et **B**. La différence par rapport au TP précédent est que l'on va réaliser les calculs en utilisant la méthode X-FEM permettant de faire passer la fissure entre les éléments du maillage. Avec cette méthode, il n'est pas nécessaire de décomposer l'éprouvette en deux *Parts* comme dans le TP précédent et il n'est pas nécessaire de supposer un trajet de fissure rectiligne. Avec cette méthode, c'est le critère d'orientation de fissure choisi qui guide la propagation de la fissure.

1 Données du problème

Géométrie La géométrie de l'éprouvette trapézoïdale étudiée est celle contenue dans le fichier *TP_trap.txt*. L'épaisseur e de l'éprouvette dans la direction \vec{e}_z prend la valeur suivante :

$$e = 10 \text{ mm.}$$

La longueur l de la pré-fissure $[GH]$ est :

$$l = 10 \text{ mm.}$$

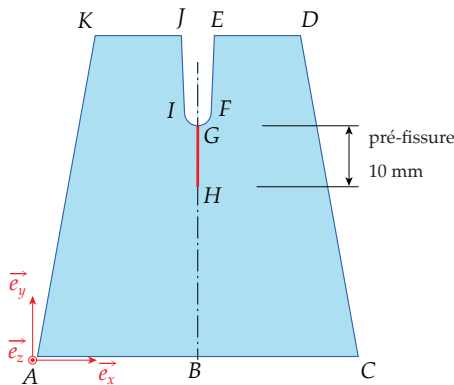


FIGURE 16.1 – Définitions des points A, B, \dots, K le long du contour de l'éprouvette

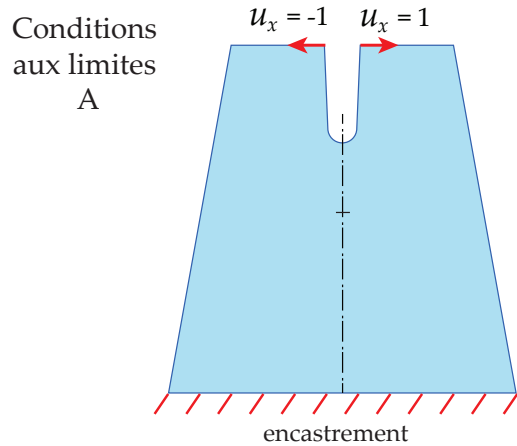


FIGURE 16.2 – Conditions aux limites **A** appliquées sur l'éprouvette

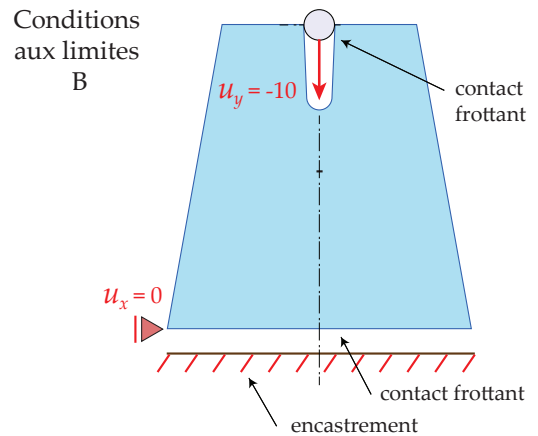


FIGURE 16.3 – Conditions aux limites **B** appliquées sur l'éprouvette.

Matériau Le matériau constituant l'éprouvette est du *Plexiglas* [4] avec les mêmes caractéristiques que dans le TP précédent. On a donc :

$$E = 3300 \text{ MPa,}$$

$$\nu = 0.37.$$

Pour la fissuration, on considère toujours que le taux de restitution d'énergie élastique G_c prend la valeur suivante :

$$G_c = 0.5 \text{ N.mm}^{-1}.$$

Dans le cas de la fissuration cohésive, on considère que la traction d'initiation t_0 prend la valeur suivante (normalement 72 MPa) pour faciliter la convergence des calculs :

$$t_0 = 10 \text{ MPa.}$$

Conditions aux limites Les conditions aux limites, notées A et B , sont définies de la même manière que pour le TP précédent.

2 Calculs dans *Abaqus*

2.1 Précautions à prendre avec la X-FEM

— La *X-FEM* ne marche pas avec les triangles dans *Abaqus*, il faudra donc utiliser des éléments quadrangles.

— Le maillage doit être construit de telle sorte qu'aucun nœud ne soit sur la pré-fissure.

— La méthode X-FEM fonctionne mieux avec des maillages réguliers dont les éléments sont orientés approximativement dans la même direction que la fissure.

— Si les conditions aux limites sont introduites sous la forme d'efforts concentrés, les contraintes sont localement dépendantes du maillage au voisinage de la zone où les efforts concentrés sont appliqués. Dans ces zones, les contraintes peuvent dépasser la contrainte d'initiation de la rupture sans que cela corresponde à une initiation de réelle de fissures dans la pratique.

Pour éviter l'apparition de ce type de fissures non souhaitées, il est possible de définir dans *Abaqus* une zone à l'intérieur de laquelle on autorise la fissure à se propager.

— La convergence est difficile à obtenir et le trajet de fissuration a tendance à être dépendant du maillage.

2.2 Résolution des problèmes

Élément ne cassant pas au bord du domaine Il peut arriver que les deux bords de la pré-fissure restent attachés au niveau du point où la fissure débouche sur le bord de l'éprouvette. Dans ce cas, il faut augmenter la longueur de la fissure pour qu'elle dépasse de la pièce.

Non convergence Avec la *X-FEM*, il est assez fréquent de voir apparaître le message d'erreur *Time increment required is less than the minimum specified* en lançant le *Job*. On peut alors essayer l'une des solutions suivantes :

- Augmenter le nombre maximum d'itérations dans un pas de temps, noté I_A . Pour cela, dans le module *Step*, aller dans le menu déroulant *Other* → *General Solution Controls* → *Edit* → *Step-1*. Cocher *Specify*, aller dans l'onglet *Time Incrementation*, cliquer sur *More*. Remplacer la valeur par défaut de I_A égale à 5 par 50.
- Vérifier que l'incrément minimal dans le *Step* est de 10^{-15} .

- Vérifier qu'il n'y a pas de nœud positionné exactement sur le trajet de la fissure.
- Faire un maillage plus régulier, la forme d'élément quadrangle optimale étant le carré.
- Avec la VCCT, le fait d'imposer une propagation rectiligne de la fissure en choisissant *Direction of crack growth relative to local 1-direction* → *Normal* peut également aider à améliorer la convergence.

Test 1 : VCCT et conditions aux limites A

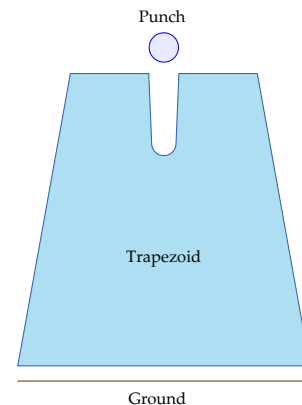


FIGURE 16.4 – a) Les 4 *Parts* définies dans *Abaqus* et b) les noms des *Sets* et des *Surfaces*

Questions

1. Importer les *Parts* *Trapezoid* et *Ground* peuvent être à partir du code JNL donné dans le fichier *TP_trap.txt* à copier-coller dans la console d'*Abaqus*. Un modèle nommé *IntraElemFrac* devrait apparaître dans l'arborescence de gauche.
2. Supprimer le modèle *Model-1*.
3. Importer le *Punch* de type *Analytical surface* en copiant les lignes correspondantes du fichier *TP_demiTrap.txt* et en les collant dans la console.
4. Renommer le modèle *IntraElemFrac* en *Model-condLimA-VCCT*.
5. Dans le module *Part* :
 - (a) Éditer le *Sketch* du *Trapezoid*. Supprimer les *Constraints* de type *Fixed* appliquées sur les points B et G . Déplacer les nœuds correspondants aux points B et G (voir sujet du TP4) de telle sorte qu'ils ne se trouvent plus sur l'axe de symétrie du *Trapezoid*. Imposer les dimensions suivantes :

$$\begin{aligned} \|\vec{BC}\| &= 10 \text{ mm,} \\ \vec{HG} \cdot \vec{u}_x &= 0.5 \text{ mm.} \end{aligned}$$

Le *Sketch* devrait ressembler à celui de la Fig. 16.5.

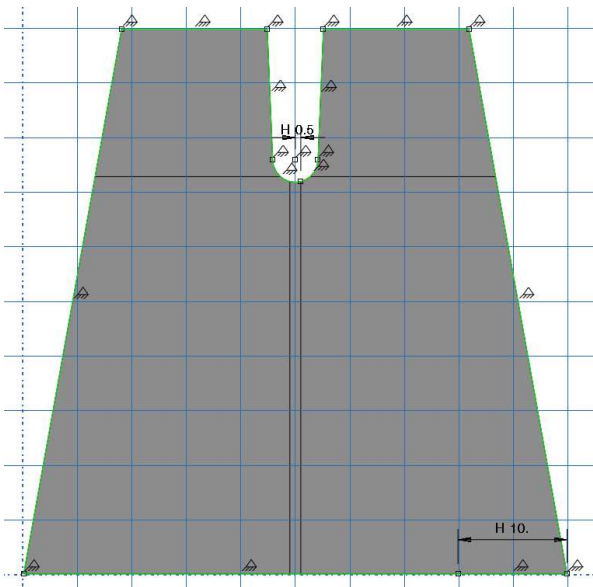


FIGURE 16.5 – Sketch du Trapezoid

- (b) En quittant le *Sketch*, un message d'erreur disant que *The feature has not been regenerated* devrait apparaître, et un symbole jaune-orangé devrait apparaître à gauche de *Shell planar-1* dans l'arborescence de gauche. Pour résoudre ce problème, faire *Clic-droit* sur *Parts* → *Trapezoid* → *Features* dans l'arborescence de gauche et sélectionner *Regenerate*.
- (c) Créer une partition de l'éprouvette trapézoïdale pour la séparer en deux parties hautes et basses conformément au schéma de la Fig. 16.9. La partie basse sera définie comme le *Crack Domain*, c'est à dire la zone à l'intérieur de laquelle la fissure a le droit de se propager. La partie haute est la zone d'application du chargement où l'on interdit à la fissure de se propager.

— Depuis le *Part* de l'éprouvette, aller dans le menu *Tools* → *Partition* et choisir *Type* → *Face* et *Method* → *Sketch*.

— Le *Sketch* de la partition devrait ressembler à celui de la Fig. 16.6.

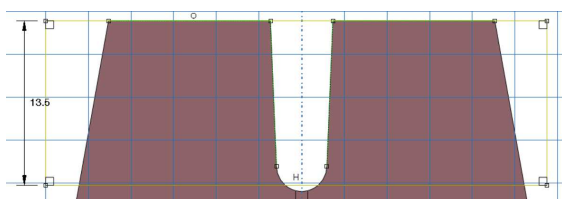


FIGURE 16.6 – Sketch de la première partition (détail)

- (d) En faisant une nouvelle *Partition*, créer une bande verticale d'épaisseur 1 mm centrée sur l'axe de symétrie de l'éprouvette.

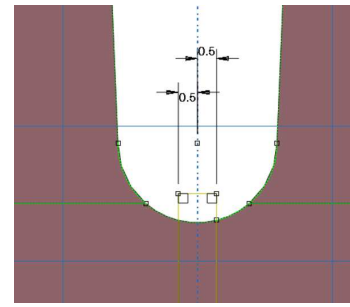


FIGURE 16.7 – Sketch de la deuxième partition (détail)

- (e) Créer un *Part* nommé *Crack* en choisissant *Modeling Space* → *2D Planar*, *Type* → *Deformable* et *Shape* → *Wire* pour la pré-fissure. La fissure étant débouchante, la longueur du *Wire* doit dépasser la longueur réelle de la fissure. On choisira une longueur de 20 mm pour le *Wire* et on fera dépasser de moitié ce *Wire* hors de l'éprouvette lors de l'assemblage. On positionnera le *Wire* de telle sorte que la fissure apparaisse directement au bon endroit lors de l'assemblage sachant que les coordonnées de *G* dans le repère global sont (25, 35.925).

6. Dans le module *Property* :

- (a) Créer un *Material* avec uniquement une partie *Mechanical* → *Elasticity* → *Elast*.
- (b) Créer la *Section* correspondante. Ne pas oublier de spécifier la bonne valeur de l'épaisseur.
- (c) Effectuer l'*Assign Section* avec la géométrie du *Trapezoid*. Il n'est pas nécessaire d'assigner une *Section* au *Part* du *Crack*.

7. Dans le module *Mesh* :

- (a) Utiliser l'outil *Mesh Controls*. Choisir *Element Shape* → *Quad*, *Technique* → *Free*, *Algorithm* → *Advancing Front*.
- (b) Spécifier une méthode d'intégration numérique permettant d'éviter les modes de déformation à énergie nulle.
- (c) Le maillage du *Part Trapezoid* est composé de quadrangles de densité globale 2.5 mm à l'exception des surfaces en contact avec le *Punch* et du contour de la bande créée pour la fissure qui sont maillées avec une densité de 1 mm (voir Fig. 16.9).
- (d) Le maillage obtenu devrait ressembler à celui de la Fig. 16.10.

8. Dans le module *Assembly* :

- (a) Créer une instance du *Part Trapezoid*.
- (b) Créer une instance du *Part Crack*.
- (c) Si besoin, utiliser l'outil *Translate Instance* pour placer correctement la pré-fissure par rapport à l'éprouvette.

9. Dans le module *Interaction* :

- (a) Créer une *Interaction Properties* nommée *IntProp-VCCT* de type *Contact* → *Mechanical* → *Fracture Criterion*. Choisir les paramètres suivants :
- Type : *VCCT*.
 - Mixed mode behavior : *BK*.
 - Critical energy release rate :
 - Mode I : 0.5 N.mm^{-1} ,
 - Mode II : 0.5 N.mm^{-1} ,
 - Mode III : 0.5 N.mm^{-1} .
 - Exponent n : 1.
- (b) Créer un *Crack* de type *X-FEM* :
- Dans menu déroulant supérieur, sélectionner *Special* → *Crack* → *Create*.
 - Sélectionner le *Crack domain* délimitant la zone à l'intérieur de laquelle on autorise la fissure à se propager conformément au dessin de la Fig. 16.9b.
 - Cocher la case *Crack location*. Le *Crack location* désigne une fissure pré-existante au calcul numérique, ici le segment $[GH]$. Sélectionner l'intégralité du *Part Crack* ayant été créé pour la pré-fissure. Cette pré-fissure est nécessaire si on utilise la *X-FEM* avec la méthode *VCCT*, car la théorie de Griffith ne prédit pas *a priori* l'initiation des fissures.
 - Cocher la case *Specify contact property*, permettant de définir une loi de propagation de fissure par l'intermédiaire d'une *Interaction Property*. Sélectionner l'*Interaction Property IntProp-VCCT* ayant été créée.
 - Le *Crack* créé devrait apparaître dans l'arborescence de gauche en suivant le chemin suivant : *Assembly* → *Engineering Features* → *Cracks*.
- (c) Créer une *Interaction* pour la X-FEM nommée *IntCrack* :
- Choisir *Step* → *Initial* et dans la fenêtre *Types for Selected Step* choisir *XFEM crack growth*.
 - Dans la fenêtre *Edit Interaction*, sélectionner le *Crack Crack-1* ayant été créé et cocher la case *Allow crack growth in this step*.
10. Dans le module *Step* :
- (a) Définir le *Step* pour qu'il y ait au moins 100 pas de temps en choisissant dans l'onglet *Incrementation* :
- Type → *Automatic*.
 - Initial increment size : 0.001.
 - Minimum increment size : $1\text{E-}15$.
 - Maximum increment size : 0.01.
- (b) Dans *Field Output Requests*, éditer *F-Output-1*.
- Cocher les cases *Failure/Fracture* → *PHILSM*, *Level set value phi* et *PSILSM*, *Level set value psi* pour que la fissure apparaisse lors de l'affichage des résultats.

11. Dans le module *Load* :

- (a) Imposer les conditions aux limites *A*.

12. Dans le module *Job* :

- (a) Créer un *Job* nommé *Job-CondLimA-VCCT*
- (b) Lancer le *Job*. Un avertissement devrait apparaître car le *Crack* n'a pas de *Section*, cependant cela n'empêche pas de lancer le calcul.
- (c) Le calcul risque de ne pas converger jusqu'à la fin du chargement imposé. On pourra utiliser les méthodes indiquées dans la section 2.2 (résolution des problèmes) pour aller plus loin dans les calcul. Notamment, on pourra augmenter la valeur du paramètre I_A spécifiant le nombre d'itérations maximal pour un pas de temps. On pourra se contenter d'une fissure se propageant jusqu'à la moitié de son chemin avant la rupture complète de l'éprouvette.

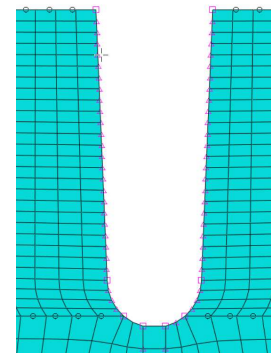


FIGURE 16.8 – Zone raffinée du maillage (détail)

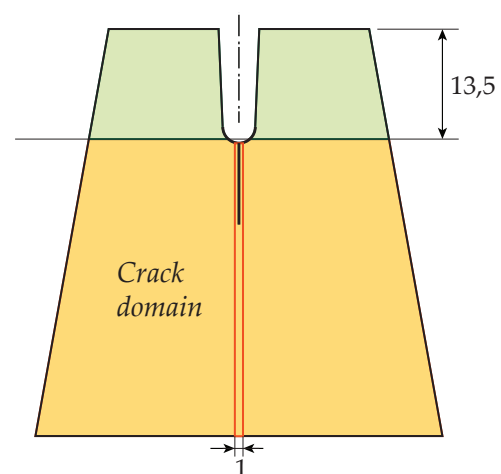
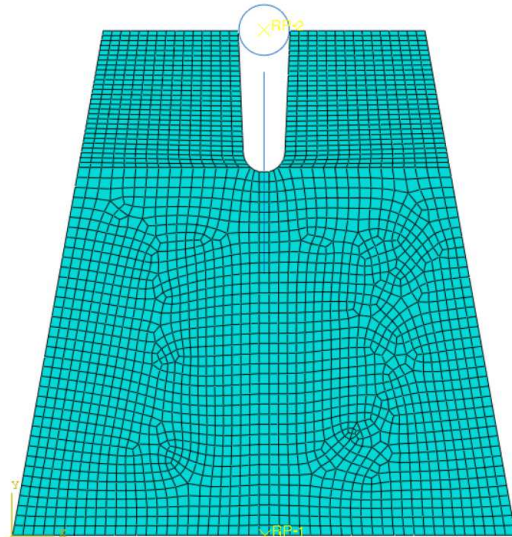


FIGURE 16.9 – Partitionnement de l'éprouvette trapézoïdale pré-fissurée

FIGURE 16.10 – Maillage du *Trapezoid* et assemblage

Test 2 : Zone cohésive et conditions aux limites A

On va maintenant voir comment modifier le test précédent pour remplacer la fissure de type VCCT par une fissure de type cohésive.

Questions

- Dupliquer le modèle *Model-condLimA-VCCT*. Renommer le nouveau modèle *Model-condLimA-CZM-1*.
- Dans le module *Material* :
 - Éditer le *Material* ayant déjà été créé pour le *Trapezoid*.
 - Créer dans la fenêtre *Material behaviors* un *Mechanical* → *Damage for Traction Separation Laws* → *Maxps Damage*.
 - Spécifier le *Max Principal Stress*.
 - Dans *Suboptions*, sélectionner *Damage Evolution*. Choisir *Type* → *Energy* et spécifier la *Fracture Energy*.
- Dans le module *Interaction* :
 - Dans l'arborescence de gauche, éditer la fissure accessible depuis le chemin *Assembly* → *Engineering Features* → *Crack*.
 - Décocher la case *Specify Contact Property*. Du fait que cette case n'est plus cochée, *Abaqus* utilisera la loi cohésive définie dans le *Material*.
- Dans le module *Job* :
 - Créer un *Job* nommé *Job-condLimA-CZM-2*.
 - Lancer un calcul.

Test 3 : Zone cohésive et conditions aux limites A (méthode alternative)

Le test 3 permet de retrouver les résultats du test 2 en utilisant une méthode alternative. La loi cohésive n'est alors plus définie dans le *Material* mais dans le module *Interaction*.

Questions

- Dupliquer le modèle *Model-condLimA-VCCT*. Renommer le nouveau modèle *Model-condLimA-CZM-2*.
 - Dans le module *Interaction* :
 - Créer une *Interaction Property* nommée *IntProp-CZM* de type *Contact* :
 - Dans la fenêtre *Contact Property Options* ajouter un *Mechanical* → *Cohesive Behavior*. Cocher la case *Allow cohesive behavior during repeated post-failure contacts*.
 - Ajouter également un *Mechanical* → *Damage* :
 - Donner la valeur de la contrainte d'initiation dans les cases *Normal Only*, *Shear-1 Only* et *Shear-2 Only*.
 - Cocher la case *Specify damage evolution*.
 - Dans l'onglet *Evolution*, choisir *Type* → *Energy* et indiquer la valeur de la *Fracture Energy*.
 - Éditer le *Crack* :
 - Vérifier que la case *Specify Contact Property* est cochée.
 - Dans le menu déroulant, sélectionner *IntProp-CZM*.
 - Dans le module *Job* :
 - Créer un *Job* nommé *Job-condLimA-CZM-2*.
 - Lancer un calcul.
- ## Test 4 : VCCT et conditions aux limites B
- Dupliquer le modèle *Model-condLimA-VCCT*. Renommer le nouveau modèle *Model-condLimB-VCCT*.
 - Dans le module *Interaction* :
 - Placer les contacts appropriés au niveau du *Ground* et du *Punch*.
 - Dans le module *Step* :
 - Éditer les *Field Output Request*, ajouter *Contact* → *CFT*, *Total forces due to contact pressure and frictional stress*.
 - Dans le module *Load* :
 - Remplacer les conditions aux limites *A* par les conditions aux limites *B*.
 - Dans le module *Visualization* :
 - Tracer la courbe de la force appliquée par le *Trapezoid* sur le *Punch* en fonction du déplacement du *Punch*.

Test 5 : Modèle cohésif et conditions aux limites B

Questions

1. Dupliquer le modèle *Model-conLimB-VCCT*. Renommer le nouveau modèle *Model-conLimB-CZM*.
2. Remplacer la fissure de type *VCCT* par une fissure de type cohésive. On s'inspirera au choix du test 2 ou du test 3.
3. Dans le module *Visualization* :
 - (a) Tracer la courbe de la force appliquée par le *Trapezoid* sur le *Punch* en fonction du déplacement du *Punch*.

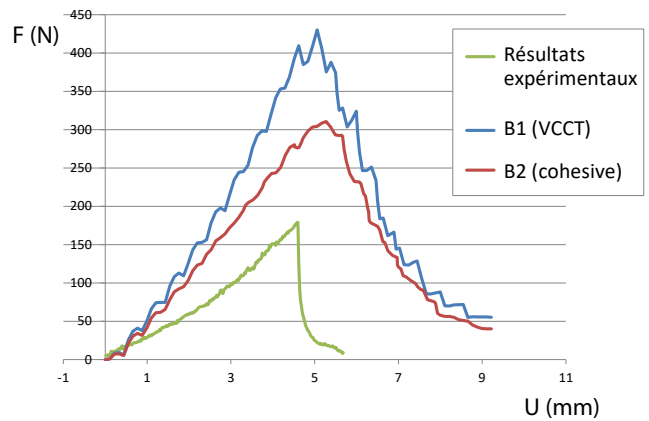


FIGURE 16.12 – Tests 4 et 5 : Courbe force-déplacement. Pour le test B2 la contrainte cohésive max est choisie égale à 10 MPa, $f = 0.45$, densités du maillage de 1 et 0.5 mm

3 Corrigé

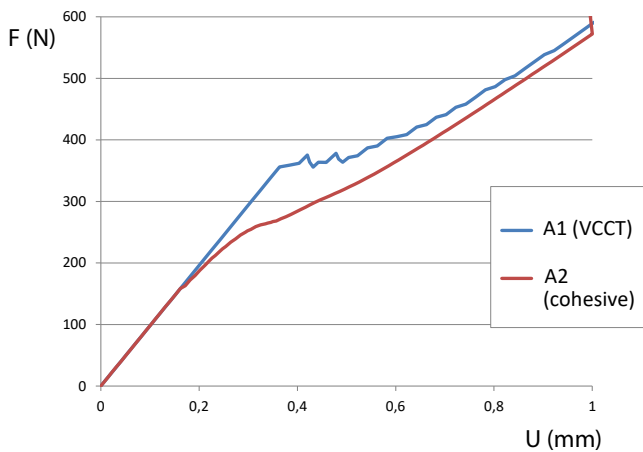


FIGURE 16.11 – Tests 1 et 2 : Courbe force-déplacement. Pour le test A2 la contrainte cohésive max est choisie égale à 10 MPa, $f = 0.45$, densités du maillage de 1 et 0.5 mm

Bibliographie

- [1] Abaqus 6.14. Abaqus/CAE User's Guide. Liens : [pdf](#) et [html](#).
- [2] Abaqus 6.14. Analysis User's Guide. Liens : [html](#), pdf : [vol. 1](#), [vol. 2](#), [vol. 3](#), [vol. 4](#), [vol. 5](#).
- [3] A. Dieye. Étude de la fissuration du plexiglas : Recherche de configurations stables et étude de l'influence des défauts. Stage de Master 2. Université Paris 13. Soutenu le 14/09/2016.
- [4] Plexiglas® GS/XT. Notice du fabricant EVONIK. Lien : [pdf](#).
- [5] J.R. Reeder. 3D Mixed-Mode Delamination Fracture Criteria - An Experimentalist's Perspective.
- [6] M.L. Benzeggagh, M. Kenane. Measurement of mixed-mode delamination fracture toughness of unidirectional glass/epoxy composites with mixed-mode bending apparatus.
- [7] P.J.G. Schreurs. Fracture Mechanics. Eindhoven University of Technology, 2012.

Annexe : Code JNL Trapèze entier

```
from part import *
from material import *
from section import *
from assembly import *
from step import *
from interaction import *
from load import *
from mesh import *
from optimization import *
from job import *
from sketch import *
from visualization import *
from connectorBehavior import *
mdb.Model( modelType = STANDARD_EXPLICIT, name = 'IntraElemFrac')
model1 = mdb.models[ 'IntraElemFrac' ]
#
# CEATION OF PART Trapezoid
#
model1.ConstrainedSketch( name = 'sketch1', sheetSize = 200.0)
# KEY
sketch1 = model1.sketches[ 'sketch1' ]
# 1
sketch1.Line( point1 = (25, 0.075), point2 = (49.9102946852, 0.075))
sketch1.FixedConstraint( entity = sketch1.vertices[ 0 ])
sketch1.FixedConstraint( entity = sketch1.vertices[ 1 ])
sketch1.FixedConstraint( entity = sketch1.geometry[2])
# 2
sketch1.Line( point1 = (49.9102946852, 0.075), point2 = (40.9372946852, 49.925))
sketch1.FixedConstraint( entity = sketch1.vertices[ 2 ])
sketch1.FixedConstraint( entity = sketch1.geometry[3])
# 3
sketch1.Line( point1 = (40.9372946852, 49.925), point2 = (27.5719400759, 49.925))
sketch1.FixedConstraint( entity = sketch1.vertices[ 3 ])
sketch1.FixedConstraint( entity = sketch1.geometry[4])
# 4
sketch1.Line( point1 = (27.5719400759, 49.925), point2 = (27.0749994014, 37.9984238101))
sketch1.FixedConstraint( entity = sketch1.vertices[ 4 ])
sketch1.FixedConstraint( entity = sketch1.geometry[5])
# 5
sketch1.ArcByCenterEnds(center = (25, 38), direction = CLOCKWISE, point1 = (27.0749994014, 37.9984238101),
point2 = (25, 35.925))
sketch1.FixedConstraint( entity = sketch1.vertices[ 5 ])
sketch1.FixedConstraint( entity = sketch1.geometry[6])
# 6
sketch1.ArcByCenterEnds(center = (25, 38), direction = CLOCKWISE, point1 = (25, 35.925), point2 =
(22.9250005986, 37.9984238101))
sketch1.FixedConstraint( entity = sketch1.vertices[ 6 ])
sketch1.FixedConstraint( entity = sketch1.geometry[7])
# 7
sketch1.Line( point1 = (22.9250005986, 37.9984238101), point2 = (22.4280599241, 49.925))
sketch1.FixedConstraint( entity = sketch1.vertices[ 7 ])
sketch1.FixedConstraint( entity = sketch1.geometry[8])
# 8
sketch1.Line( point1 = (22.4280599241, 49.925), point2 = (9.06270531478, 49.925))
sketch1.FixedConstraint( entity = sketch1.vertices[ 8 ])
sketch1.FixedConstraint( entity = sketch1.geometry[9])
# 9
sketch1.Line( point1 = (9.06270531478, 49.925), point2 = (0.0897053147753, 0.075))
sketch1.FixedConstraint( entity = sketch1.vertices[ 9 ])
sketch1.FixedConstraint( entity = sketch1.geometry[10])
```

```
# 10
sketch1.Line( point1 = (0.0897053147753, 0.075), point2 = (25, 0.075))
sketch1.FixedConstraint( entity = sketch1.vertices[ 10 ])
sketch1.FixedConstraint( entity = sketch1.geometry[11])
#
model1.Part( dimensionality = TWO_D_PLANAR, name = 'Trapezoid', type = DEFORMABLE_BODY)
# KEY
part1 = model1.parts[ 'Trapezoid' ]
#
part1.BaseShell( sketch = sketch1)
del sketch1
#
# CEATION OF PART Ground
#
model1.ConstrainedSketch( name = 'sketch2', sheetSize = 200.0)
# KEY
sketch2 = model1.sketches[ 'sketch2' ]
# 1
sketch2.Line( point1 = (0, 0.075), point2 = (25, 0.075))
sketch2.FixedConstraint( entity = sketch2.vertices[ 0 ])
sketch2.FixedConstraint( entity = sketch2.vertices[ 1 ])
sketch2.FixedConstraint( entity = sketch2.geometry[2])
# 2
sketch2.Line( point1 = (25, 0.075), point2 = (50, 0.075))
sketch2.FixedConstraint( entity = sketch2.vertices[ 2 ])
sketch2.FixedConstraint( entity = sketch2.geometry[3])
#
model1.Part( dimensionality = TWO_D_PLANAR, name = 'Ground', type = ANALYTIC_RIGID_SURFACE)
# KEY
part2 = model1.parts[ 'Ground' ]
#
part2.AnalyticRigidSurf2DPlanar( sketch = sketch2)
del sketch2
```

Annexe : Code JNL Demi-Trapèze

```
from part import *
from material import *
from section import *
from assembly import *
from step import *
from interaction import *
from load import *
from mesh import *
from optimization import *
from job import *
from sketch import *
from visualization import *
from connectorBehavior import *
mdb.Model( modelType = STANDARD_EXPLICIT, name = 'InterElemFrac')
model1 = mdb.models[ 'InterElemFrac' ]
#
# CEATION OF PART HalfTrapezoid-Rigth
#
model1.ConstrainedSketch( name = 'sketch1', sheetSize = 200.0)
# KEY
sketch1 = model1.sketches[ 'sketch1' ]
# 1
sketch1.Line( point1 = (25, 0.075), point2 = (49.9102946852, 0.075))
sketch1.FixedConstraint( entity = sketch1.vertices[ 0 ])
sketch1.FixedConstraint( entity = sketch1.vertices[ 1 ])
sketch1.FixedConstraint( entity = sketch1.geometry[2])
# 2
sketch1.Line( point1 = (49.9102946852, 0.075), point2 = (40.9372946852, 49.925))
sketch1.FixedConstraint( entity = sketch1.vertices[ 2 ])
sketch1.FixedConstraint( entity = sketch1.geometry[3])
# 3
sketch1.Line( point1 = (40.9372946852, 49.925), point2 = (27.5719400759, 49.925))
sketch1.FixedConstraint( entity = sketch1.vertices[ 3 ])
sketch1.FixedConstraint( entity = sketch1.geometry[4])
# 4
sketch1.Line( point1 = (27.5719400759, 49.925), point2 = (27.0749994014, 37.9984238101))
sketch1.FixedConstraint( entity = sketch1.vertices[ 4 ])
sketch1.FixedConstraint( entity = sketch1.geometry[5])
# 5
sketch1.ArcByCenterEnds(center = (25, 38), direction = CLOCKWISE, point1 = (27.0749994014, 37.9984238101),
point2 = (25, 35.925))
sketch1.FixedConstraint( entity = sketch1.vertices[ 5 ])
sketch1.FixedConstraint( entity = sketch1.geometry[6])
# 6
sketch1.Line( point1 = (25, 35.925), point2 = (25, 26))
sketch1.FixedConstraint( entity = sketch1.vertices[ 6 ])
sketch1.FixedConstraint( entity = sketch1.geometry[7])
# 7
sketch1.Line( point1 = (25, 26), point2 = (25, 0.075))
sketch1.FixedConstraint( entity = sketch1.vertices[ 7 ])
sketch1.FixedConstraint( entity = sketch1.geometry[8])
#
model1.Part( dimensionality = TWO_D_PLANAR, name = 'HalfTrapezoid-Rigth', type = DEFORMABLE_BODY)
# KEY
part1 = model1.parts[ 'HalfTrapezoid-Rigth' ]
#
part1.BaseShell( sketch = sketch1)
del mdb.models[ 'InterElemFrac' ].sketches[ 'sketch1' ]
#
# ADDING SETS AND SURFACES
```

```

#
part1.Set(edges = part1.edges.findAt( ((37.4551473426, 0.075, 0.0)), name = 'Set-Bottom-Right')
part1.Surface( name = 'Surf-Bottom-Right', side1Edges = part1.edges.findAt( ((37.4551473426, 0.075, 0.0))))
part1.Set(edges = part1.edges.findAt( ((27.3234697386, 43.9617119051, 0.0)), name = 'Set-Notch-Right')
part1.Surface( name = 'Surf-Notch-Right', side1Edges = part1.edges.findAt( ((27.3234697386, 43.9617119051,
0.0))))
part1.Set(edges = part1.edges.findAt( ((25, 30.9625, 0.0)), name = 'Set-CrackIni-Right')
part1.Surface( name = 'Surf-CrackIni-Right', side1Edges = part1.edges.findAt( ((25, 30.9625, 0.0))))
part1.Set(edges = part1.edges.findAt( ((25, 13.0375, 0.0)), name = 'Set-CrackCalc-Right')
part1.Surface( name = 'Surf-CrackCalc-Right', side1Edges = part1.edges.findAt( ((25, 13.0375, 0.0))))
part1.Set(edges = part1.edges.findAt( ((25, 30.9625, 0.0), (25, 13.0375, 0.0)), name =
'Set-CrackTot-Right')
part1.Surface( name = 'Surf-CrackTot-Right', side1Edges = part1.edges.findAt( ((25, 30.9625, 0.0), (25,
13.0375, 0.0))))
#
# CEATION OF PART HalfTrapezoid-Left
#
model1.ConstrainedSketch( name = 'sketch2', sheetSize = 200.0)
# KEY
sketch2 = model1.sketches[ 'sketch2' ]
# 1
sketch2.Line( point1 = (25, 0.075), point2 = (0.0897053147753, 0.075))
sketch2.FixedConstraint( entity = sketch2.vertices[ 0 ])
sketch2.FixedConstraint( entity = sketch2.vertices[ 1 ])
sketch2.FixedConstraint( entity = sketch2.geometry[2])
# 2
sketch2.Line( point1 = (0.0897053147753, 0.075), point2 = (9.06270531478, 49.925))
sketch2.FixedConstraint( entity = sketch2.vertices[ 2 ])
sketch2.FixedConstraint( entity = sketch2.geometry[3])
# 3
sketch2.Line( point1 = (9.06270531478, 49.925), point2 = (22.4280599241, 49.925))
sketch2.FixedConstraint( entity = sketch2.vertices[ 3 ])
sketch2.FixedConstraint( entity = sketch2.geometry[4])
# 4
sketch2.Line( point1 = (22.4280599241, 49.925), point2 = (22.9250005986, 37.9984238101))
sketch2.FixedConstraint( entity = sketch2.vertices[ 4 ])
sketch2.FixedConstraint( entity = sketch2.geometry[5])
# 5
sketch2.ArcByCenterEnds(center = (25, 38), direction = COUNTERCLOCKWISE, point1 = (22.9250005986,
37.9984238101), point2 = (25, 35.925))
sketch2.FixedConstraint( entity = sketch2.vertices[ 5 ])
sketch2.FixedConstraint( entity = sketch2.geometry[6])
# 6
sketch2.Line( point1 = (25, 35.925), point2 = (25, 26))
sketch2.FixedConstraint( entity = sketch2.vertices[ 6 ])
sketch2.FixedConstraint( entity = sketch2.geometry[7])
# 7
sketch2.Line( point1 = (25, 26), point2 = (25, 0.075))
sketch2.FixedConstraint( entity = sketch2.vertices[ 7 ])
sketch2.FixedConstraint( entity = sketch2.geometry[8])
#
model1.Part( dimensionality = TWO_D_PLANAR, name = 'HalfTrapezoid-Left', type = DEFORMABLE_BODY)
# KEY
part2 = model1.parts[ 'HalfTrapezoid-Left' ]
#
part2.BaseShell( sketch = sketch2)
del mdb.models[ 'InterElemFrac' ].sketches[ 'sketch2' ]
#
# ADDING SETS AND SURFACES
#
part2.Set(edges = part2.edges.findAt( ((12.5448526574, 0.075, 0.0)), name = 'Set-Bottom-Left')
part2.Surface( name = 'Surf-Bottom-Left', side1Edges = part2.edges.findAt( ((12.5448526574, 0.075, 0.0))))
part2.Set(edges = part2.edges.findAt( ((22.6765302614, 43.9617119051, 0.0)), name = 'Set-Notch-Left')
part2.Surface( name = 'Surf-Notch-Left', side1Edges = part2.edges.findAt( ((22.6765302614, 43.9617119051,
0.0))))
part2.Set(edges = part2.edges.findAt( ((25, 30.9625, 0.0)), name = 'Set-CrackIni-Left')
part2.Surface( name = 'Surf-CrackIni-Left', side1Edges = part2.edges.findAt( ((25, 30.9625, 0.0))))
part2.Set(edges = part2.edges.findAt( ((25, 13.0375, 0.0)), name = 'Set-CrackCalc-Left')
part2.Surface( name = 'Surf-CrackCalc-Left', side1Edges = part2.edges.findAt( ((25, 13.0375, 0.0))))
part2.Set(edges = part2.edges.findAt( ((25, 30.9625, 0.0), (25, 13.0375, 0.0)), name =
'Set-CrackTot-Left')
part2.Surface( name = 'Surf-CrackTot-Left', side1Edges = part2.edges.findAt( ((25, 30.9625, 0.0), (25,
13.0375, 0.0))))
#

```

```

# CEATION OF PART Punch
#
model1.ConstrainedSketch( name = 'sketch3', sheetSize = 200.0)
# KEY
sketch3 = model1.sketches[ 'sketch3' ]
# 1
sketch3.ArcByCenterEnds(center = (25, 50), direction = COUNTERCLOCKWISE, point1 = (27.5, 50), point2 = (25,
52.5))
sketch3.FixedConstraint( entity = sketch3.vertices[ 0 ])
sketch3.FixedConstraint( entity = sketch3.vertices[ 1 ])
sketch3.FixedConstraint( entity = sketch3.geometry[2])
# 2
sketch3.ArcByCenterEnds(center = (25, 50), direction = COUNTERCLOCKWISE, point1 = (25, 52.5), point2 = (22.5,
50))
sketch3.FixedConstraint( entity = sketch3.vertices[ 2 ])
sketch3.FixedConstraint( entity = sketch3.geometry[3])
# 3
sketch3.ArcByCenterEnds(center = (25, 50), direction = COUNTERCLOCKWISE, point1 = (22.5, 50), point2 = (25,
47.5))
sketch3.FixedConstraint( entity = sketch3.vertices[ 3 ])
sketch3.FixedConstraint( entity = sketch3.geometry[4])
# 4
sketch3.ArcByCenterEnds(center = (25, 50), direction = COUNTERCLOCKWISE, point1 = (25, 47.5), point2 = (27.5,
50))
sketch3.FixedConstraint( entity = sketch3.vertices[ 4 ])
sketch3.FixedConstraint( entity =sketch3.geometry[5])
#
model1.Part( dimensionality = TWO_D_PLANAR, name = 'Punch', type = ANALYTIC_RIGID_SURFACE)
# KEY
part3 = model1.parts[ 'Punch' ]
#
part3.AnalyticRigidSurf2DPlanar( sketch = sketch3)
del mdb.models[ 'InterElemFrac' ].sketches[ 'sketch3' ]
#
# CEATION OF PART Ground
#
model1.ConstrainedSketch( name = 'sketch4', sheetSize = 200.0)
# KEY
sketch4 = model1.sketches[ 'sketch4' ]
# 1
sketch4.Line( point1 = (0, 0.075), point2 = (25, 0.075))
sketch4.FixedConstraint( entity = sketch4.vertices[ 0 ])
sketch4.FixedConstraint( entity = sketch4.vertices[ 1 ])
sketch4.FixedConstraint( entity = sketch4.geometry[2])
# 2
sketch4.Line( point1 = (25, 0.075), point2 = (50, 0.075))
sketch4.FixedConstraint( entity = sketch4.vertices[ 2 ])
sketch4.FixedConstraint( entity = sketch4.geometry[3])
#
model1.Part( dimensionality = TWO_D_PLANAR, name = 'Ground', type = ANALYTIC_RIGID_SURFACE)
# KEY
part4 = model1.parts[ 'Ground' ]
#
part4.AnalyticRigidSurf2DPlanar( sketch = sketch4)
del mdb.models[ 'InterElemFrac' ].sketches[ 'sketch4' ]

```