

ZZZZ189 – Orientation circonférentielle par une boucle Python

Résumé :

On traite le cas d'un hémisphère en béton avec armatures orthoradiales sous pression en élasticité linéaire. On cherche à orienter circonférentiellement les armatures. Cette opération peut être réalisée de deux manières.

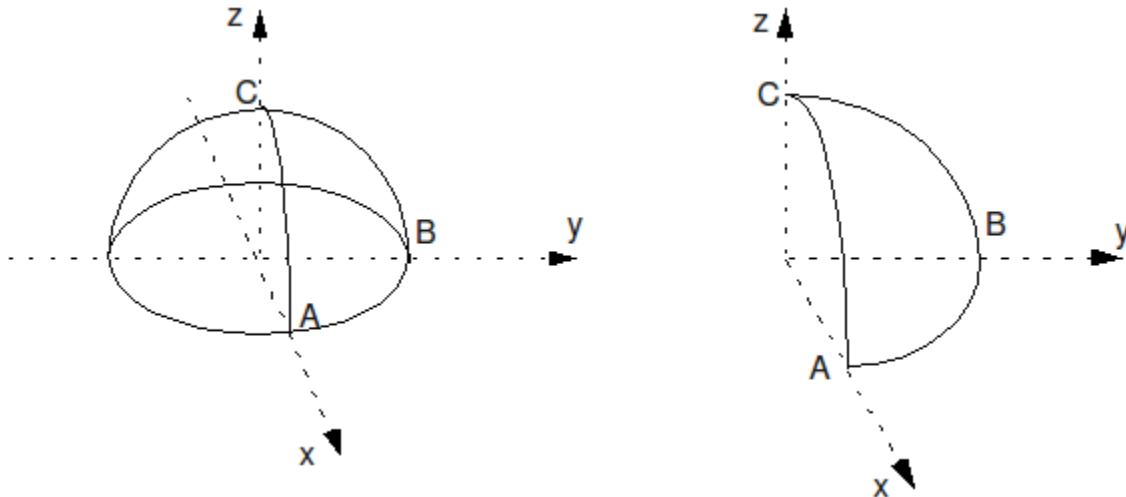
On propose dans les modélisations (A) et (C) une fonction Python qui permet de réaliser cette action, en orientant les mailles une par une.

Dans les modélisations (B) et (D), on utilise la commande `AXE` de l'option `GRILLE` de `AFFE_CARA_ELEM` qui permet une mise en donnée bien plus simple du fichier de commande pour orienter toutes les armatures avec une seule commande. Les résultats de la modélisation (B) sont ensuite comparés à la modélisation (A), et ceux de la modélisation (D) à la modélisation (C).

Dans les modélisations (A) et (B), l'axe de référence est $(0,0,1)$. Dans les modélisations (C) et (D), l'axe de référence est $(0,1,0)$. Les deux dernières modélisations ont été ajoutées afin d'avoir des valeurs d'angle nautique β non nulles.

1 Problème de référence

1.1 Géométrie



Rayon $R = 10. m$
Épaisseur $t = 0.04$

Coordonnées des points :

	A	B	C
x	10.	0.	0.
y	0.	10.	0.
z	0.	0.	10.

1.2 Propriétés de matériaux

L'hémisphère est composé de béton renforcé par des grilles. Comme l'objectif est uniquement de tester des commandes, on donne les mêmes propriétés au béton et aux grilles.

$E = 200000 Pa$, $\nu = 0.3$

1.3 Conditions aux limites et chargements

Sur un quart de l'hémisphère :

Point C pas de déplacement en z
Côté AC symétrie par rapport au plan xz
Côté BC symétrie par rapport au plan yz
Côté AB libre

Pression interne : $P = 10. Pa$

1.4 Problématique d'orientation

On veut que les grilles de renforcement soient orientées circonférentiellement par rapport à l'axe donné. Or, vue la forme hémisphérique considérée, on propose une fonction Python pour le réaliser.

Le principe est le suivant :

1) l'axe de l'hémisphère est noté Z

- 2) pour chaque maille, on calcule le vecteur indiquant la direction ciconférentielle par $V = Z \wedge N$ où N est la normale à la maille. On note $V = [V_x, V_y, V_z]$ et on définit sa projection sur le plan (XOY) par $W = [V_x, V_y, 0]$
- 3) on calcule les valeurs de ANGL_REP à affecter à la maille courante par $\alpha = \arctan(V_y/V_x)$ et $\beta = -\arctan(V_z/\text{norm}(W))$

La fonction appelée LIST_CARA_CIRCONF boucle sur toutes les mailles grille, calcule α et β et crée les mots-clé :

```
_F(SECTION=20.0,  
  MAILLE=Nom_Maille_Courante,  
  EXCENTREMENT=0.0, ANGL_REP=( $\alpha$ ,  $\beta$ )),  
  COEF_RIGI_DRZ=1.E-10  
) ,
```

et il les ajoute dans une liste. On fournit ensuite cette liste à AFFE_CARA_ELEM. On obtient donc les commandes :

```
LIST_GRI=LIST_CARA_CIRCONF(  
  GROUP_MA='GRILLE',  
  AXE=(0.,0.,1.),  
  MODELE=MODEL,  
  GRILLE=_F(SECTION=20., EXCENTREMENT=0.,)  
)  
  
CARA_COQ=AFFE_CARA_ELEM(  
  MODELE=MODEL,  
  COEF_RIGI_DRZ=0.,),  
  GRILLE=LIST_GRI,  
)
```

Notons enfin que, formellement, LIST_CARA_CIRCONF obéit au catalogue suivant :

```
LIST_CARA_CIRCONF(  
  GROUP_MA           =SIMP(statut='o', typ=grma, max='**'),  
  AXE                =SIMP(statut='o', typ='R', max=3, min=3),  
  MODE               =SIMP(statut='o', typ=modele_sdaster),  
  GRILLE             =FACT(statut='f', max=1,  
    SECTION          =SIMP(statut='o', typ='R'),  
    EXCENTREMENT     =SIMP(statut='f', typ='R'),  
    COEF_RIGI_DRZ     =SIMP(statut='f', typ='R'),  
  ),  
)
```

2 Modélisation A

2.1 Caractéristiques de la modélisation

1373 éléments de coque DKT

1373 éléments de grille GRILLE_MEMBRANE

Modélisation d'un quart de l'hémisphère en TRIA3.

2.2 Caractéristiques du maillage

Nombre de nœuds : 734

Nombre de mailles et types : 2746 TRIA3

2.3 Grandeurs testées et résultats

On teste des valeurs de non-régression calculées avec la version V7.03.30.

Identification	Type de référence	Valeurs de référence
Nœud 30, déplacement <i>DX</i>	'NON_REGRESSION'	0
Nœud 30, déplacement <i>DY</i>	'NON_REGRESSION'	3.139397E-05
Nœud 30, déplacement <i>DZ</i>	'NON_REGRESSION'	1.533531E-05
Nœud 700, déplacement <i>DX</i>	'NON_REGRESSION'	4.487504E-06
Nœud 700, déplacement <i>DY</i>	'NON_REGRESSION'	3.321124E-05
Nœud 700, déplacement <i>DZ</i>	'NON_REGRESSION'	1.515258E-05

3 Modélisation B

La modélisation B teste la fonctionnalité `AXE` de l'option `GRILLE de AFPE_CARA_ELEM` qui permet d'orienter les armatures directement par Code_Aster. Les résultats utilisant les boucles python servent à fournir la solution de référence.

3.1 Caractéristiques de la modélisation

1373 éléments de coque `DKT`
1373 éléments de grille `GRILLE_MEMBRANE`

Modélisation d'un quart de l'hémisphère en `TRIA3`.

3.2 Caractéristiques du maillage

Nombre de nœuds : 734
Nombre de mailles et types : 2746 `TRIA3`

3.3 Grandeurs testées et résultats

On compare les valeurs de déplacement avec ceux calculés par la boucle Python (modélisation `A`).

Identification	Type de référence	Valeurs de référence
Nœud 30, déplacement <code>DX</code>	'NON_REGRESSION'	0
Nœud 30, déplacement <code>DY</code>	'NON_REGRESSION'	3.139397E-05
Nœud 30, déplacement <code>DZ</code>	'NON_REGRESSION'	1.533531E-05
Nœud 700, déplacement <code>DX</code>	'NON_REGRESSION'	4.487504E-06
Nœud 700, déplacement <code>DY</code>	'NON_REGRESSION'	3.321124E-05
Nœud 700, déplacement <code>DZ</code>	'NON_REGRESSION'	1.515258E-05

4 Modélisation C

4.1 Caractéristiques de la modélisation

1373 éléments de coque DKT

1373 éléments de grille GRILLE_MEMBRANE

Modélisation d'un quart de l'hémisphère en TRIA3.

4.2 Caractéristiques du maillage

Nombre de nœuds : 734

Nombre de mailles et types : 2746 TRIA3

4.3 Grandeurs testées et résultats

On teste des valeurs de non-régression calculées avec la version VII.02.20.

Identification	Type de référence	Valeurs de référence
Nœud 30, déplacement <i>DX</i>	'NON_REGRESSION'	0
Nœud 30, déplacement <i>DY</i>	'NON_REGRESSION'	3.8636255E-05
Nœud 30, déplacement <i>DZ</i>	'NON_REGRESSION'	1.4680190E-05
Nœud 700, déplacement <i>DX</i>	'NON_REGRESSION'	3.9129862E-06
Nœud 700, déplacement <i>DY</i>	'NON_REGRESSION'	4.0527685E-05
Nœud 700, déplacement <i>DZ</i>	'NON_REGRESSION'	1.4931766E-05

5 Modélisation D

La modélisation D teste la fonctionnalité `AXE` de l'option `GRILLE` de `APPE_CARA_ELEM` qui permet d'orienter les armatures directement par Code_Aster. Les résultats utilisant les boucles python servent à fournir la solution de référence (modélisation C).

5.1 Caractéristiques de la modélisation

1373 éléments de coque `DKT`
1373 éléments de grille `GRILLE_MEMBRANE`

Modélisation d'un quart de l'hémisphère en `TRIA3`.

5.2 Caractéristiques du maillage

Nombre de nœuds : 734
Nombre de mailles et types : 2746 `TRIA3`

5.3 Grandeurs testées et résultats

On compare les valeurs de déplacement avec ceux calculés par la boucle Python (modélisation C)

Identification	Type de référence	Valeurs de référence
Nœud 30, déplacement <i>DX</i>	'NON_REGRESSION'	0
Nœud 30, déplacement <i>DY</i>	'NON_REGRESSION'	3.8636255E-05
Nœud 30, déplacement <i>DZ</i>	'NON_REGRESSION'	1.4680190E-05
Nœud 700, déplacement <i>DX</i>	'NON_REGRESSION'	3.9129862E-06
Nœud 700, déplacement <i>DY</i>	'NON_REGRESSION'	4.0527685E-05
Nœud 700, déplacement <i>DZ</i>	'NON_REGRESSION'	1.4931766E-05

6 Synthèse des résultats

Ce test présente une utilisation avancée du langage Python au sein de *Code_Aster* (modélisations *A* et *C*) qui sert de référence à la validation des la modélisations *B* et *D*. Ces modélisations utilisent la commande `AXE` de l'option `GRILLE` de `AFFE_CARA_ELEM`. Les solutions de référence sont retrouvées.