

CONCEPTION MÉCANIQUE ASSISTÉE PAR ORDINATEUR

Modélisation dans ABAQUS/Maple

Travail de synthèse

Présenté
À
M. A. Gakwaya

Bryan Elliott-Tam

(111 133 067)



Faculté des sciences et de génie
13 Décembre 2018

Définition du problème

L'objectif du travail est de modéliser une coque de pression de sous-marin soumise à une pression de design de 12304.586 KPa (équivalent à une profondeur de 12 km). Une première modélisation est effectuée à l'aide d'un script python avec plusieurs matériaux, une seconde est effectuée en modifiant la première sur Abaqus afin de rajouter des renfort interne qui pénètre dans la coque, une troisième est effectuée avec des renfort interne qui sort de la structure et une dernière avec des renfort interne dans la structure. Par la suite une optimisation est effectuée sur Abaqus et finalement une analyse analytique est effectuée à l'aide de python.

Identification du problème

Équation au dérivé partiel d'une coque soumise à une pression.

$$a_1 = \frac{2}{N} \sum_{i=0}^{N-1} R(\theta_i) \sin \theta_i, \quad a_n = \frac{2}{N} \sum_{i=0}^{N-1} R(\theta_i) \sin n\theta_i$$

$$\frac{d^4 w}{dx^4} + \left(\frac{Pa}{2EI_c} \right) \frac{d^2 w}{dx^2} + \left(\frac{h}{I_c a^2} \right) w = \frac{P}{EI_c} \left(1 - \frac{\mu}{2} \right)$$

Référence

(Structural Analysis and Design of PressureHulls: the State of the Art and Future Trends de John R. MacKay)

Condition limite

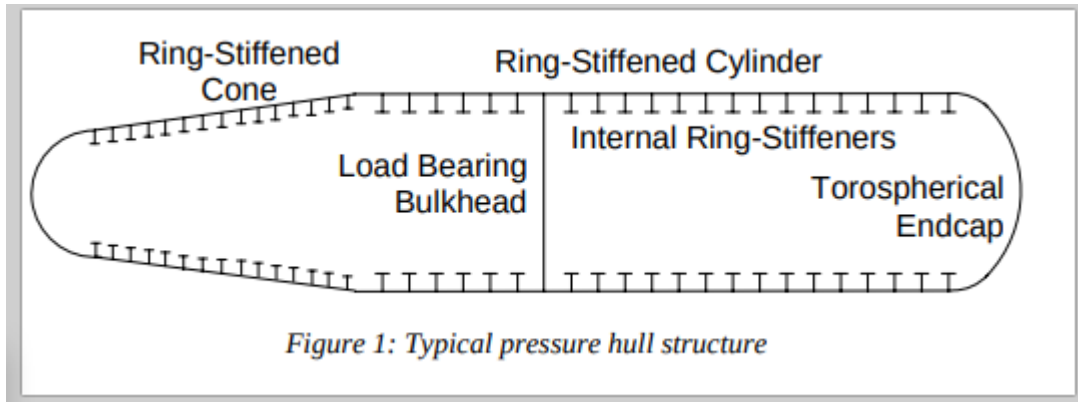
D'après Structural Analysis and Design of PressureHulls: the State of the Art and Future Trends de John R. MacKay, les conditions limites influence beaucoup les résultats par contre les condition limites de rotation on moins d'effet sur les résultats.

Chargement

Le chargement est une pression uniforme de 12304.586 Pa vers l'intérieur de la structure

Choix du modèle

Le modèle est basé sur la figure plus bas (sans les anneaux interne), car c'est la forme classique d'une coque de sous-marin. La longueur est de 154m et la hauteur de 12 à 16 m, car c'est les dimensions que plusieurs sous-marin partage



(référence : Structural Analysis and Design of PressureHulls: the State of the Art and Future Trends de John R. MacKay)

Formulation faible

$$\int_V \left[\frac{d^4 w}{dx^4} + \left(P * \frac{a}{2EI_c} \right) \left(\frac{d^2 w}{dx^2} \right) + \left(\frac{h}{I_c a^2} \right) w - \frac{P}{EI_c} \left(1 - \frac{\mu}{2} \right) \right] dV = 0$$

$$\int_V \left[\frac{d^4 w}{dx^4} \right] dV + \int_V \left[\left(P * \frac{a}{2EI_c} \right) \left(\frac{d^2 w}{dx^2} \right) + \left(\frac{h}{I_c a^2} \right) w \right] dV + \int_V \left[- \frac{P}{EI_c} \left(1 - \frac{\mu}{2} \right) \right] dV$$

Comme le nom « coque de pression » l'indique des élément coque ont été choisi, car la paroi de la coque est mince par rapport à son envergure et la rotation n'est pas un mode de déformation très significatif.

Modélisation sur Abaqus

Génération de la géométrie

Caractéristique de la modélisation

Maillage : Seed global 1
 Pression : 12304 Kpa
 Modèle : Shell 3D révolution
 Matériau : Titane, acier,aluminum
 Condition limite : aucune rotation sur tout le modèle

```

# -*- coding: utf-8 -*-
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 job import *
from sketch import *
from visualization import *
from connectorBehavior import *
import regionToolset
import assembly
import odbAccess

def main(list_materiau,differentiel_pression=12304.586*1000,epaisseur=1):
    for propriete_materiau in list_materiau:
        model_coque = mdb.models['Model-1']
        model_coque = mdb.Model(name=propriete_materiau[-1], objectToCopy=mdb.models['Model-1'])
        model_coque.ConstrainedSketch(name='__profile__', sheetSize=250.0)

        sketch = model_coque.sketches['__profile__']
        #####geometrie#####
        sketch.ArcByCenterEnds(center=(16,0),point2=(0,0),point1=(16,16))
        sketch.Line(point1=(16,16),point2=(100,16))
        sketch.Line(point1=(100,16),point2=(154-12,12))
        sketch.ArcByCenterEnds(center=(154-12,0),point2=(154-12,12),point1=(154,0))
        sketch.Line(point1=(0,0),point2=(154,0))
        sketch.assignCenterline(line=sketch.ConstructionLine(point1=(0,0),point2=(154,0)))

        #####Revolution#####
        p = model_coque.Part(name='Coque', dimensionality=THREE_D,
            type=DEFORMABLE_BODY)
        p = model_coque.parts['Coque']
        p.BaseShellRevolve(sketch=sketch, angle=360)

        #####Material et section#####
        model_coque.Material(name=propriete_materiau[-1])
        materiau = model_coque.materials[propriete_materiau[-1]]
        coefficient_poisson = propriete_materiau[1]
        young_modulus =propriete_materiau[2]
        materiau.Elastic(type=ISOTROPIC,table=((young_modulus,coefficient_poisson),))

        model_coque.HomogeneousShellSection(material=propriete_materiau[-1], name=
            'Section-1', thickness=epaisseur)

        f = p.faces

        faces_1 = f.findAt(((154, 0, 0),))
        region_1 = regionToolset.Region(faces=faces_1)
        p.SectionAssignment(region=region_1,sectionName='Section-1',offset=0.0)

        faces_2 = f.findAt(((16, 16, 0),))
        region_2 = regionToolset.Region(faces=faces_2)
        p.SectionAssignment(region=region_2,sectionName='Section-1',offset=0.0)

        faces_3 = f.findAt(((0, 0, 0),))

```

```

region_3 = regionToolset.Region(faces=faces_3)
p.SectionAssignment(region=region_3,sectionName='Section-1',offset=0.0)

faces_4 = f.findAt((121, 14, 0),)
region_4 = regionToolset.Region(faces=faces_4)
p.SectionAssignment(region=region_4,sectionName='Section-1',offset=0.0)
#####Assembly#####

model_coque.rootAssembly.Instance(name='assemblage',part=p,dependent=ON)
a = model_coque.rootAssembly
model_coque.rootAssembly.regenerate()

#####Step#####
model_coque.StaticStep(initialInc=0.1,maxInc=0.1,name='Step-1',previous='Initial')
#####Mesh#####
p.seedPart(size=1)
p.generateMesh()
#####BC#####
''' Toutes les rotations sont contrainte'''
f = a.instances['assemblage'].faces

faces_1 = f.findAt((154, 0, 0),)
region_assembly_1 = regionToolset.Region(faces=faces_1)
model_coque.DisplacementBC(name='Contrainte_rotation_1',createStepName='Step-1',region=region_assembly_1,url=0,ur2=0,ur3=0)

faces_2 = f.findAt((16, 16, 0),)
region_assembly_2 = regionToolset.Region(faces=faces_2)
model_coque.DisplacementBC(name='Contrainte_rotation_2',createStepName='Step-1',region=region_assembly_2,url=0,ur2=0,ur3=0)

faces_3 = f.findAt((0, 0, 0),)
region_assembly_3 = regionToolset.Region(faces=faces_3)
model_coque.DisplacementBC(name='Contrainte_rotation_3',createStepName='Step-1',region=region_assembly_3,url=0,ur2=0,ur3=0)

faces_4 = f.findAt((121,14,0),)
region_assembly_4 = regionToolset.Region(faces=faces_4)
model_coque.DisplacementBC(name='Contrainte_rotation_4',createStepName='Step-1',region=region_assembly_4,url=0,ur2=0,ur3=0)

#####Load#####
'''Pression appliquer partout'''

region_assembly_1 = regionToolset.Region(sidelFaces=faces_1)
model_coque.Pressure(name='Pression_1',createStepName='Step-1',region=region_assembly_1,magnitude=differentiel_pression)

region_assembly_2 = regionToolset.Region(sidelFaces=faces_2)
model_coque.Pressure(name='Pression_2',createStepName='Step-1',region=region_assembly_2,magnitude=differentiel_pression)

region_assembly_3 = regionToolset.Region(sidelFaces=faces_3)
model_coque.Pressure(name='Pression_3',createStepName='Step-1',region=region_assembly_3,magnitude=differentiel_pression)

region_assembly_4 = regionToolset.Region(sidelFaces=faces_4)
model_coque.Pressure(name='Pression_4',createStepName='Step-1',region=region_assembly_4,magnitude=differentiel_pression)

#####Job#####
a.regenerate()
job = mdb.Job(name=propiete_materiau[0],model=propiete_materiau[-1],type=ANALYSIS,description='Analyse préliminaire')
job.submit()
job.waitForCompletion()

del model_coque.sketches['__profile__']

acier = ['job_acier',0.25,200*10**9,'acier']
titane = ['job_titane',0.342,113.8*10**9,'titane']
aluminium = ['job_aluminium',0.33,68.9*10**9,'aluminum']
main([acier,titane,aluminium])

```

Résultat

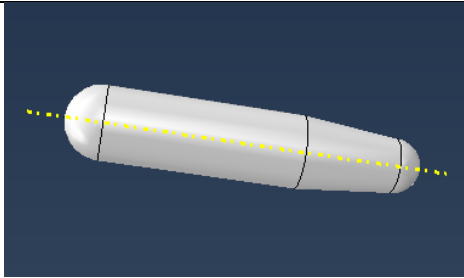


Figure 1 modèle

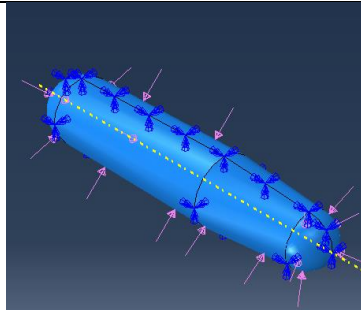


Figure 2Chargement

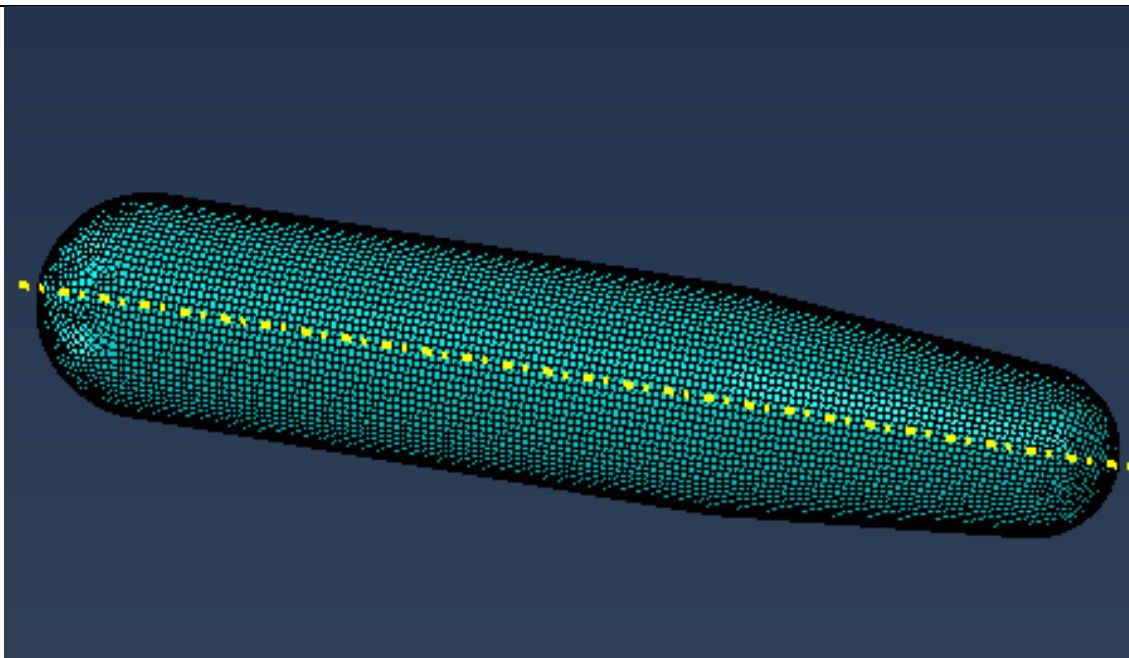


Figure 3maillage

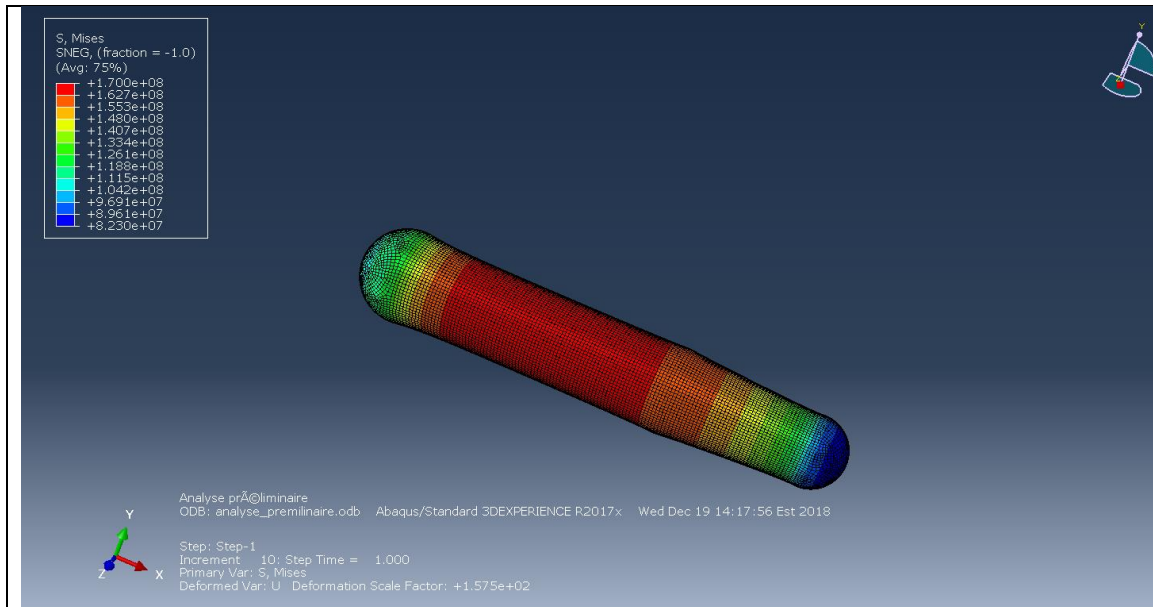


Figure 4 Contrainte de von mises titane

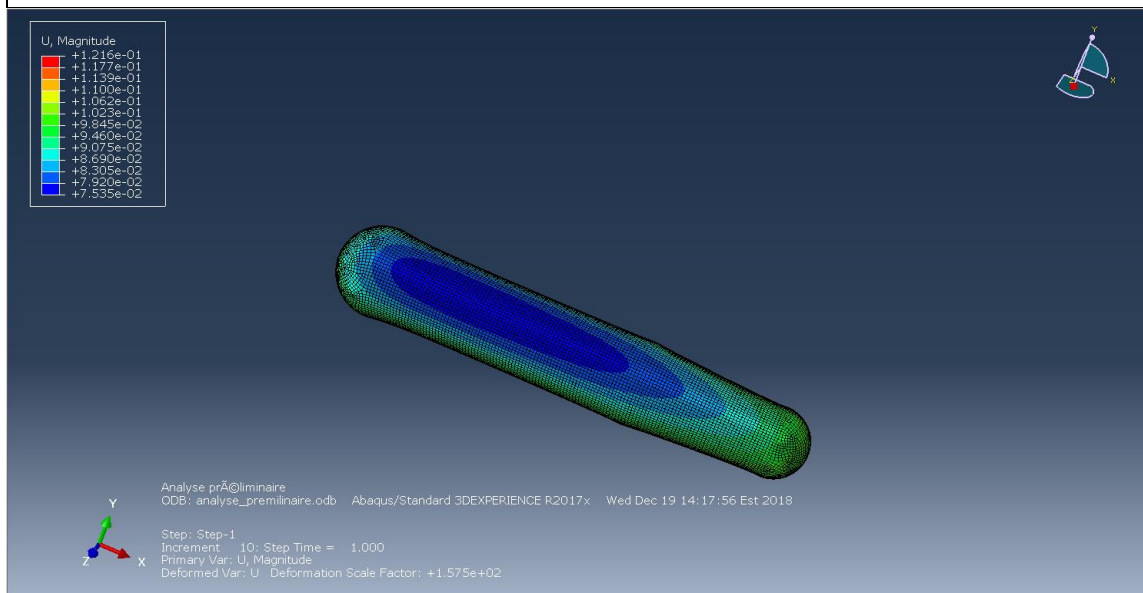


Figure 5 D placement avec une  chelle de 1.575 02 titane

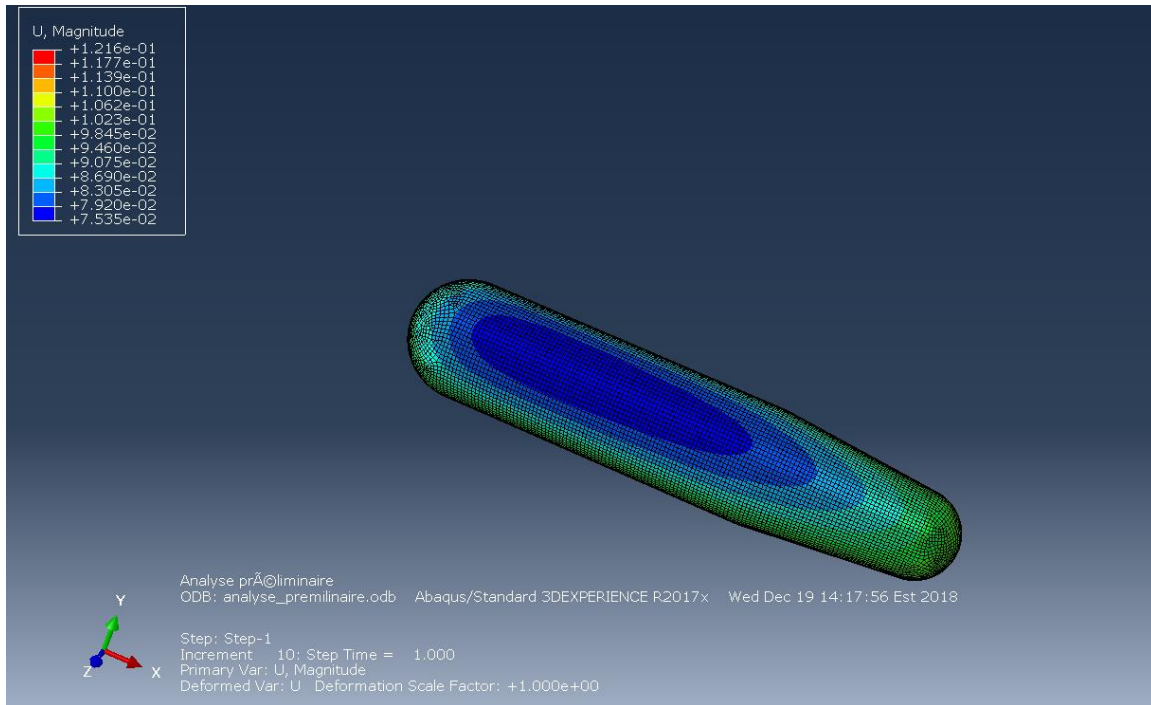


Figure 6 Déplacement avec une échelle de 1 titane

Path->Node

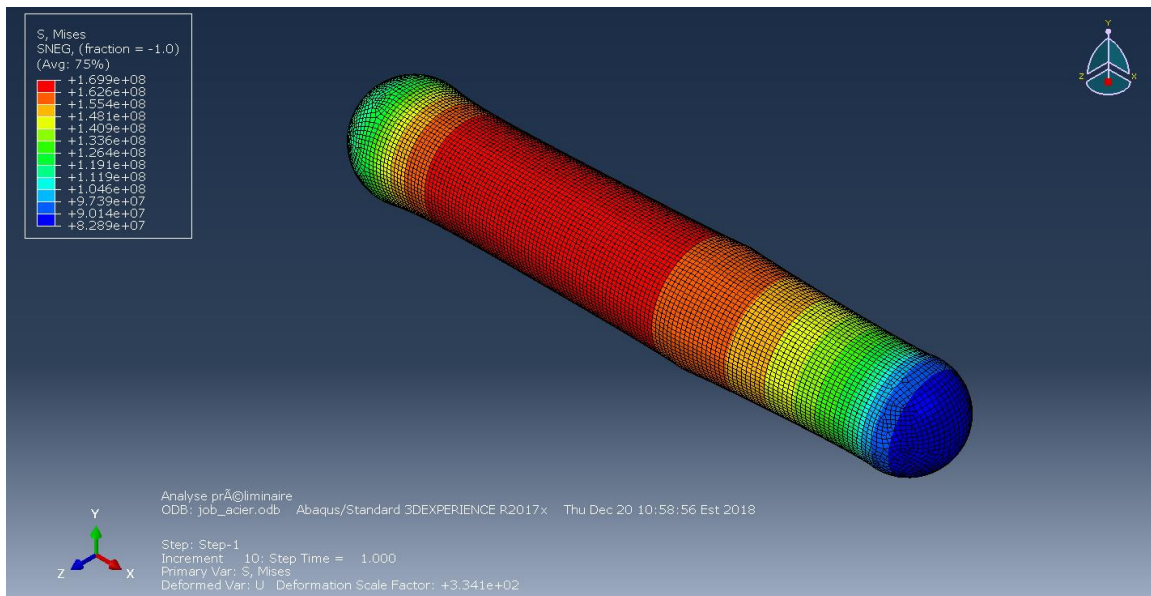


Figure 7 Von mises acier

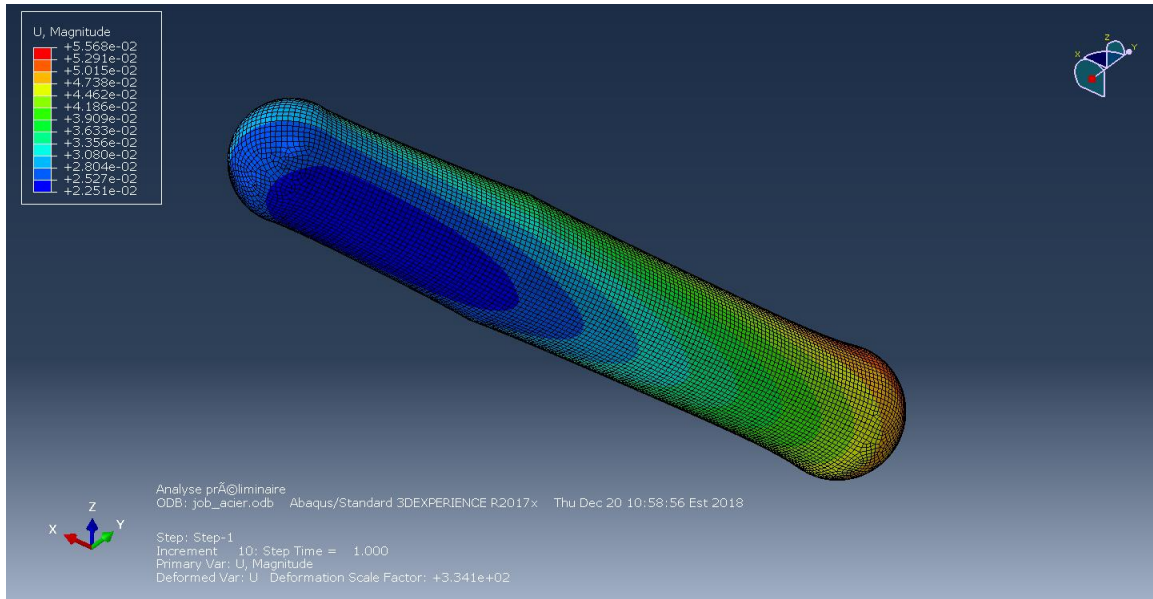


Figure 8 D placement acier

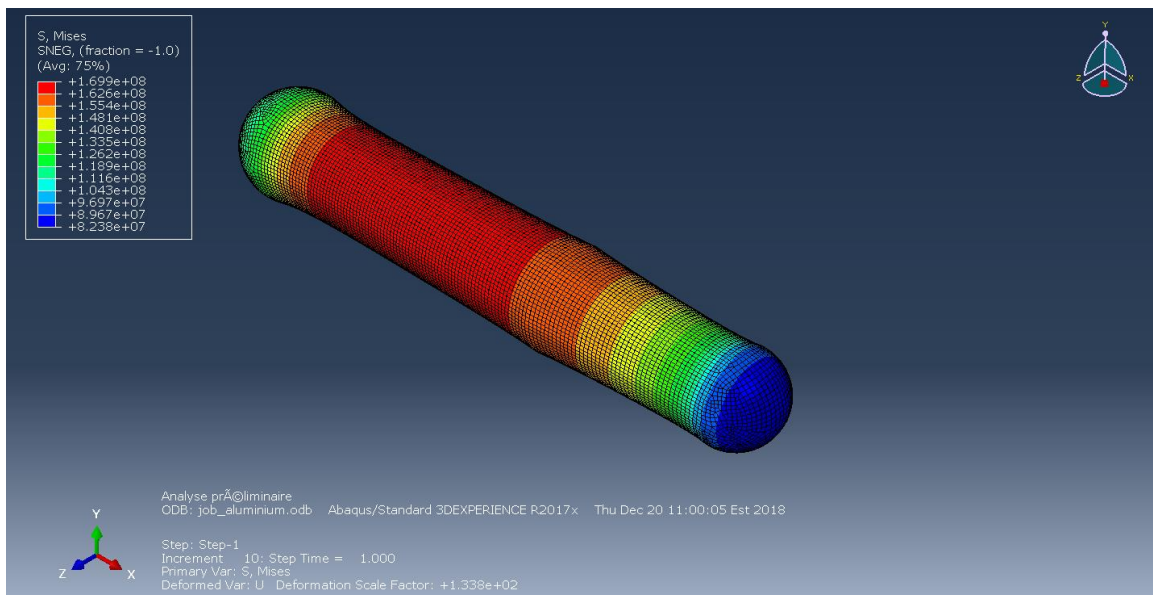


Figure 9 Von mises aluminium

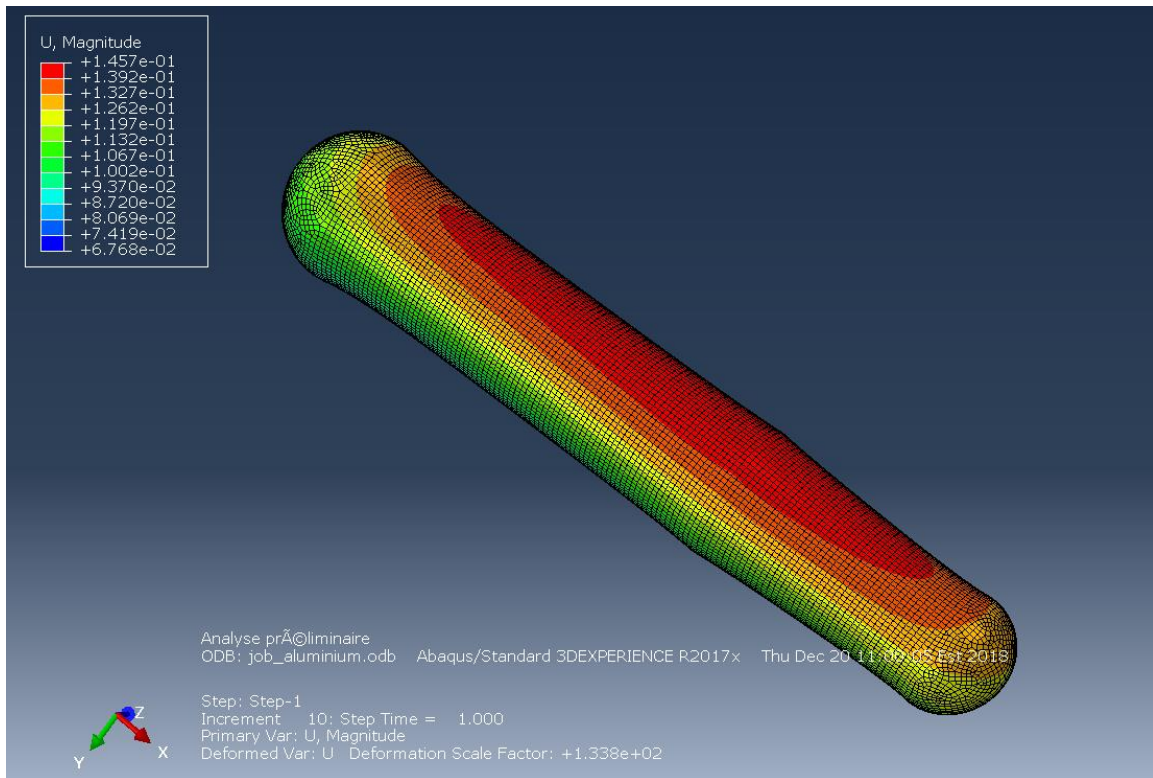


Figure 10 D placement aluminium

Analyse simplifier

Afin de v rifier les r sultats d'Abaqus une analyse tr s simplifier est effectu  en supposant un simple cylindre. Les r sultats sont du m me ordre de grandeur. Cela ne valide pas la solution d'Abaqus, mais permet d'avoir plus de confiance dans la coh rence de l'analyse.

Il est supposé que la coque est un cylindre, d'un rayon de 16m avec un épaisseur de 1m

```
In [22]: import math
from decimal import Decimal

In [29]: p = 12304.586*1000
t=1
r=16
l =154

In [30]: sigmax = p*r/t

In [31]: sigmay=p*r/2*t

In [32]: print('sigmax: '+ '%.3E' % Decimal(str(sigmax)))
print('sigmay: '+ '%.3E' % Decimal(str(sigmay)))

sigmax: 1.969E+08
sigmay: 9.844E+07

In [33]: von_mises = (1/math.sqrt(2))*(((sigmax-sigmay)**2)+(sigmay**2)+(sigmax**2))**(1/2)
print('Von mises : '+ '%.3E' % Decimal(str(von_mises)))

Von mises : 1.705E+08

In [34]: materiaux = [['acier',200*10**9,0.25],['titane',113.8*10**9,0.342],['aluminum',68.9*10**9,0.33]]
for materiau in materiaux:
    delta_r = (von_mises/materiau[1])*r
    delta_l = delta_r*materiau[2]*l
    print('le déplacement radial du {} est de {} m et le déplacement longitudinal est de {} m'.format(materiau[0],delta_r,delta_l))

le déplacement radial du acier est de 0.013639787595584447 m et le déplacement longitudinal est de 0.525131822430
0012 m
le déplacement radial du titane est de 0.023971507197863704 m et le déplacement longitudinal est de 1.26253134109
70858 m
le déplacement radial du aluminum est de 0.039592997374700865 m et le déplacement longitudinal est de 2.012116126
582298 m
```

Discussion

On voit telle qu'attendus le matériau avec le plus grand module d'élasticité a le plus petit déplacement et la contrainte de Von mises ne change pas beaucoup entre les matériaux. Tous les matériaux ne feront pas défaillance avec cette pression.

Matériau	Contrainte maximal	Limite élastique
Acier	1.69e+8	3.5e+8
Aluminium	1.699e+8	2.76e+8
Titane	1.7e+8	8.8e+8

Renforcement de la coque

Cette section modifier le modèle précédant

Premier Renfort

Copy model ->edit sketch -> ajout renfort ->répétition linéaire -> supprimer les excédents



Figure 11 Renfort

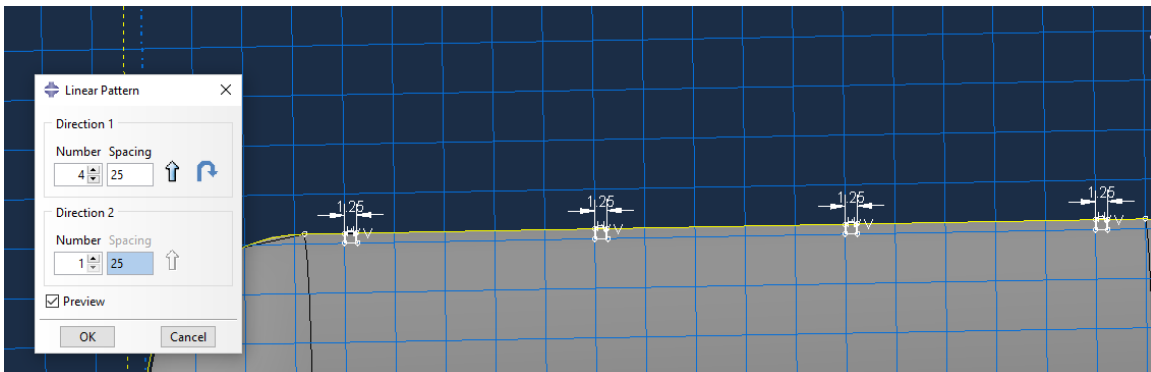


Figure 12 répétition linéaire

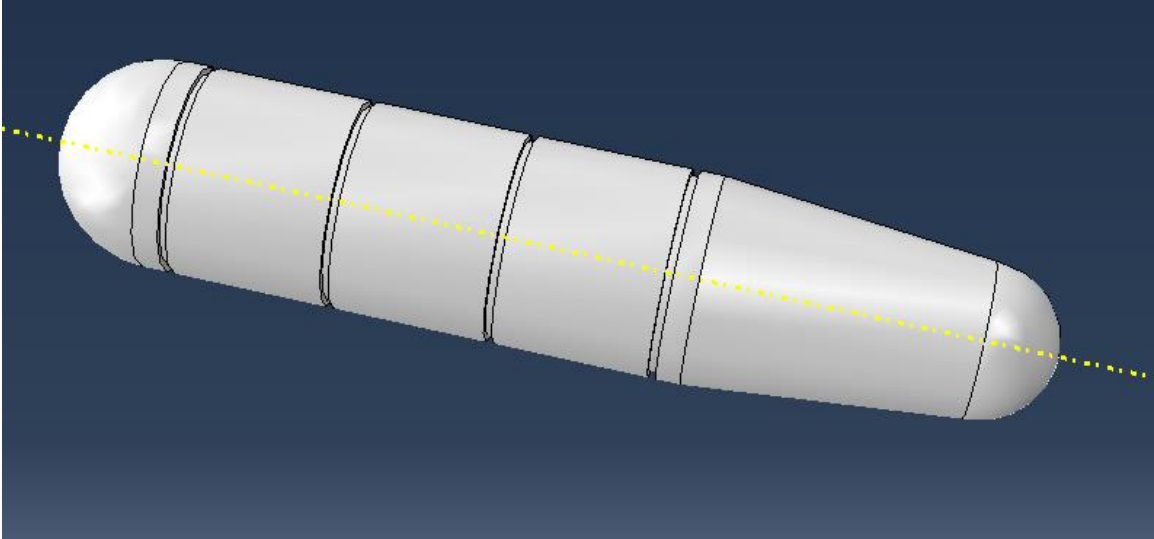


Figure 13 nouveau modèle

Section Assignments -> supprimer ->refaire avec tous le modèle

Mesh->seed global :1 ->mesh part

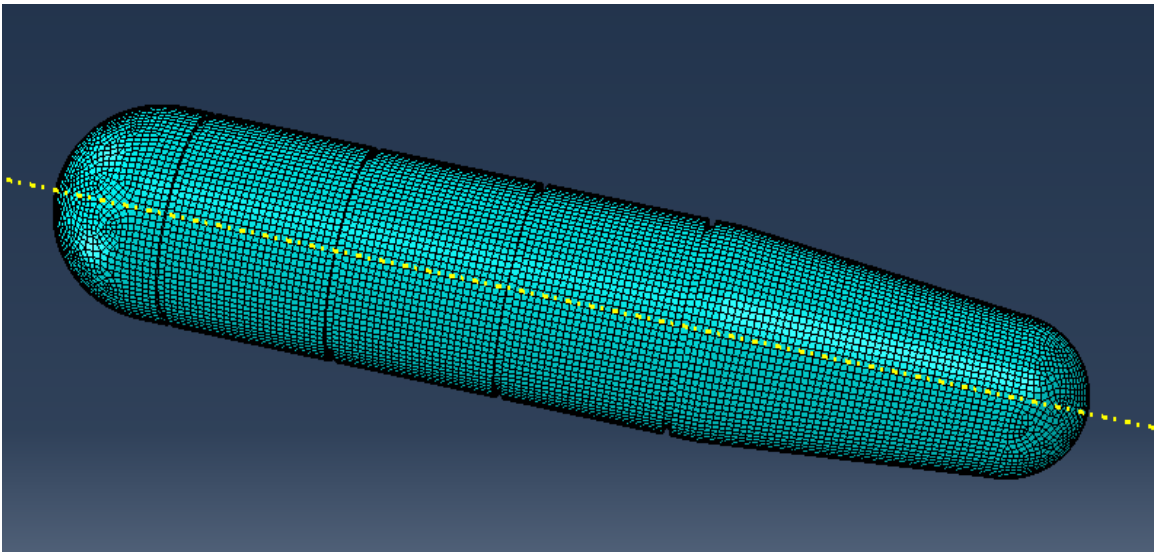


Figure 14 Maillage

Supprimer les conditions limites et BC et les refaire en sélectionnant tous le modèle

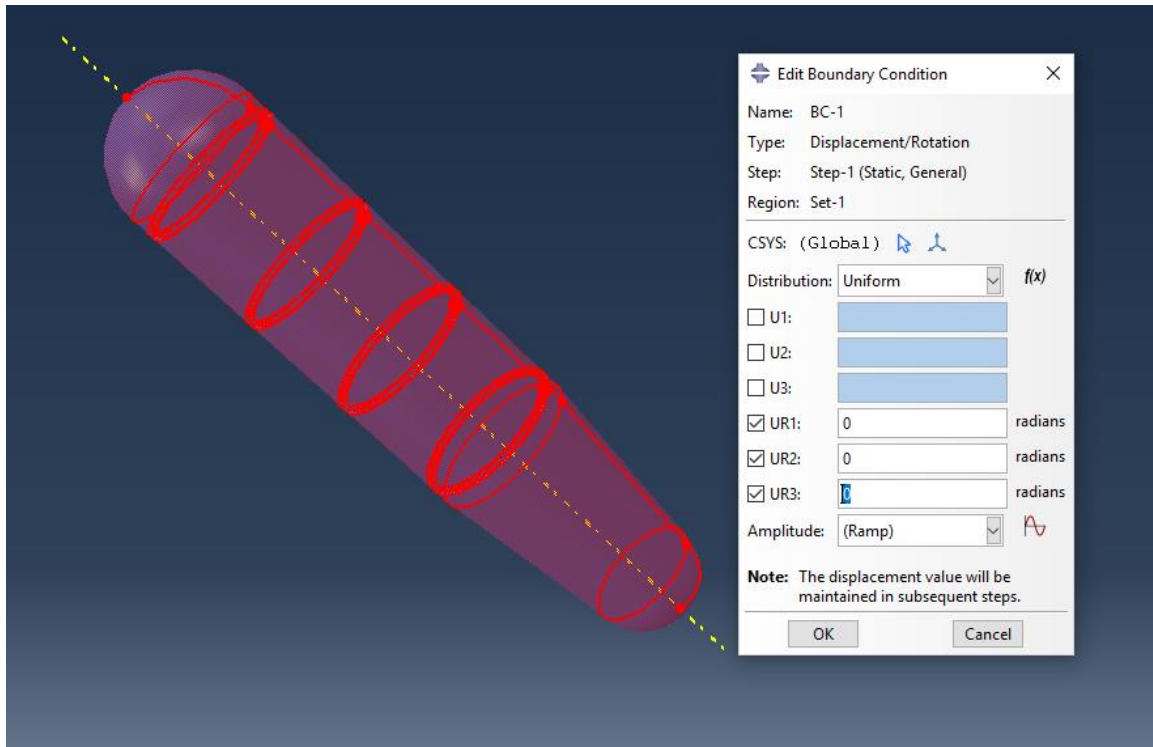


Figure 15 Conditions limites

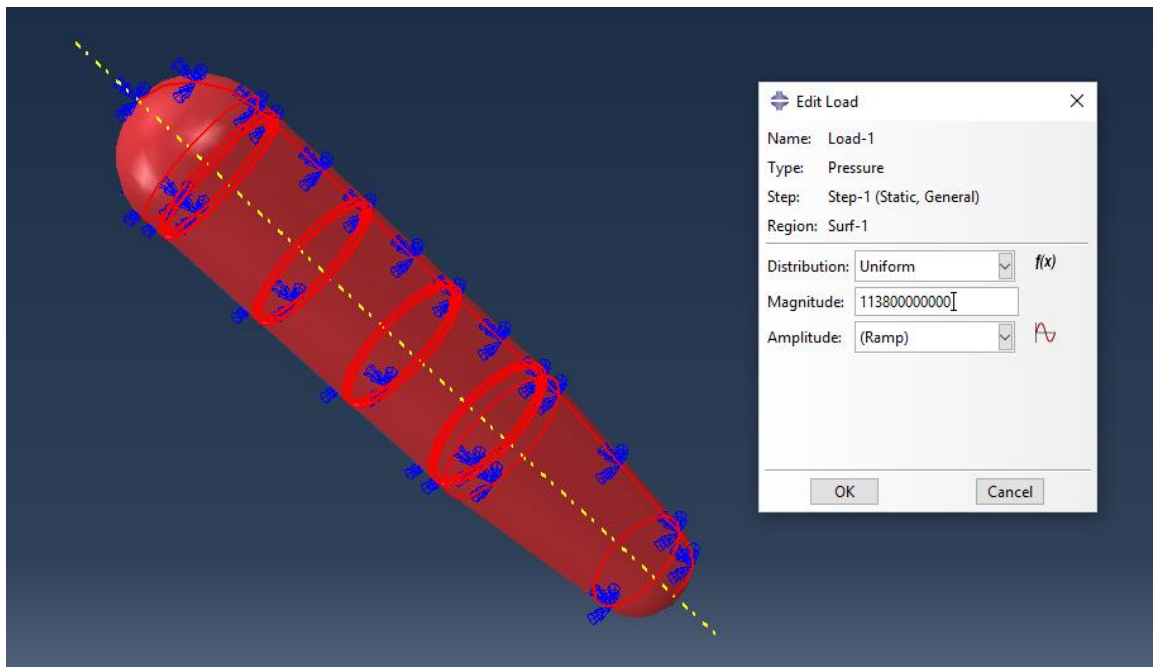


Figure 16 Chargement

Résultat

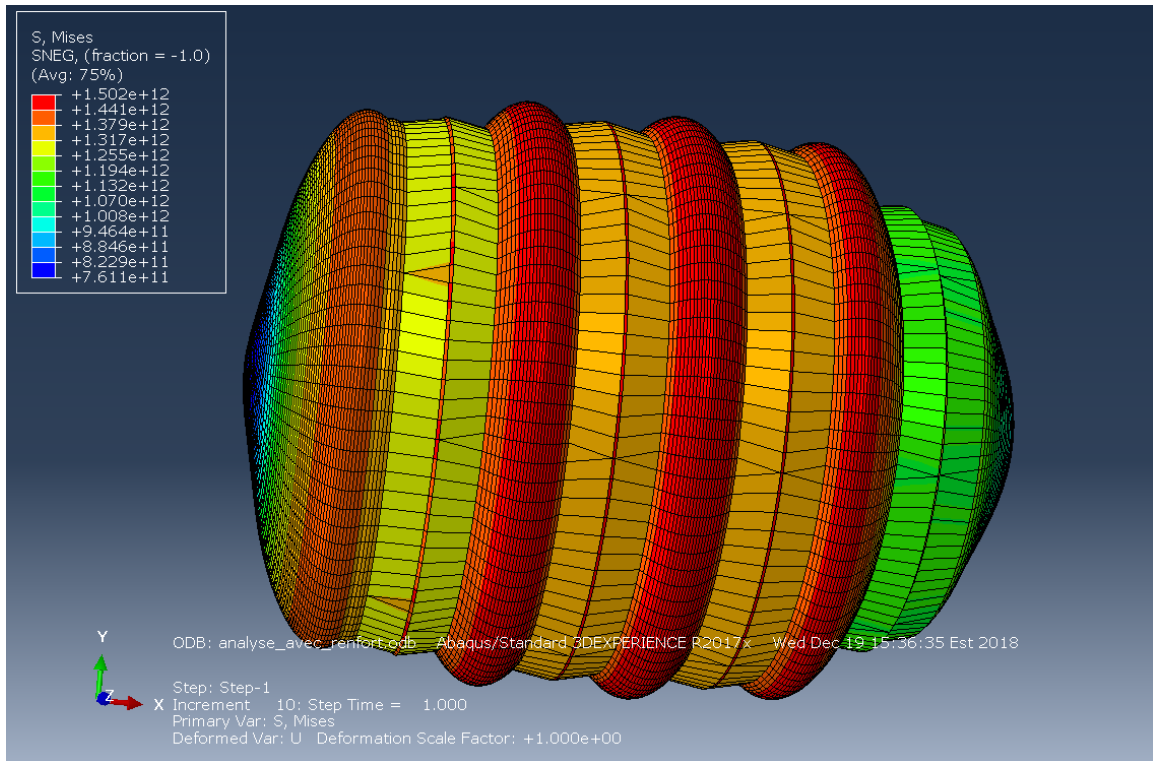


Figure 17 Contrainte

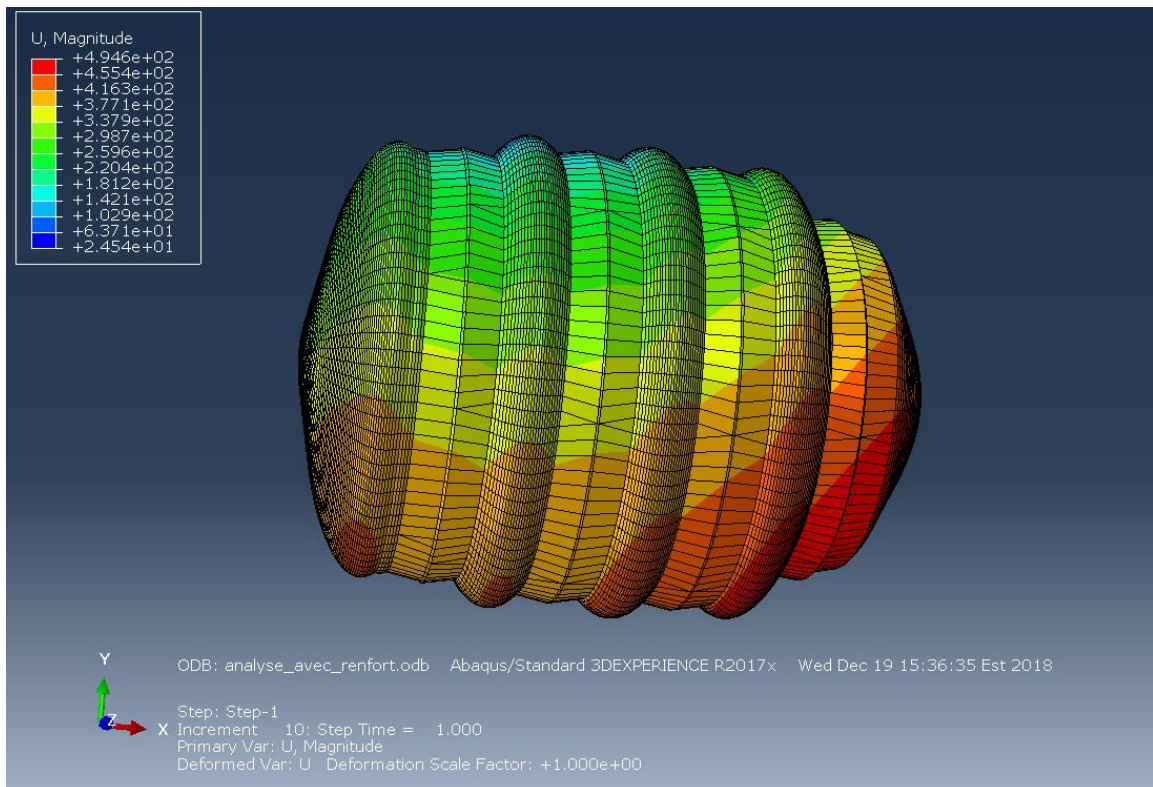


Figure 18 Déplacement

Deuxième renfort

Copy model -> edit sketch -> ajouté renfort -> répétition -> enlever les excédants

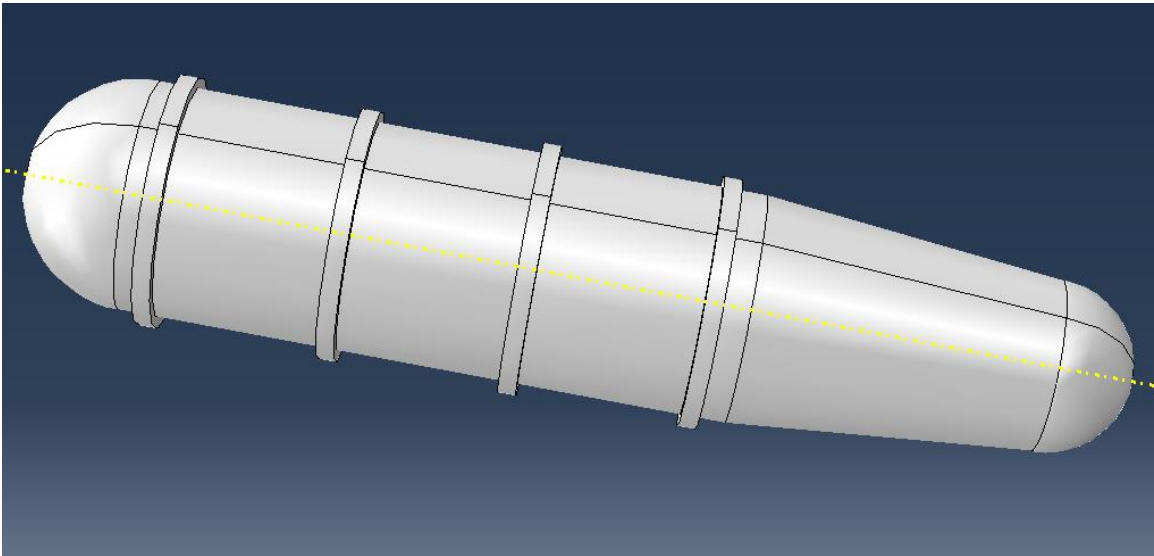


Figure 19 modèle

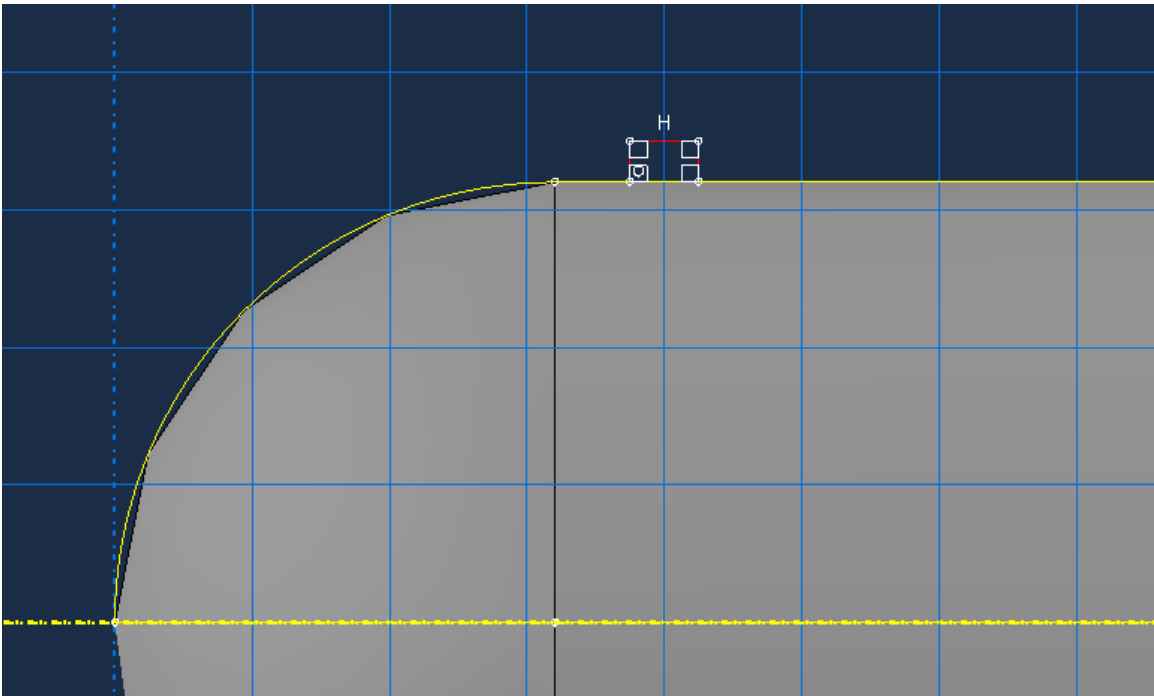


Figure 20 Renfort

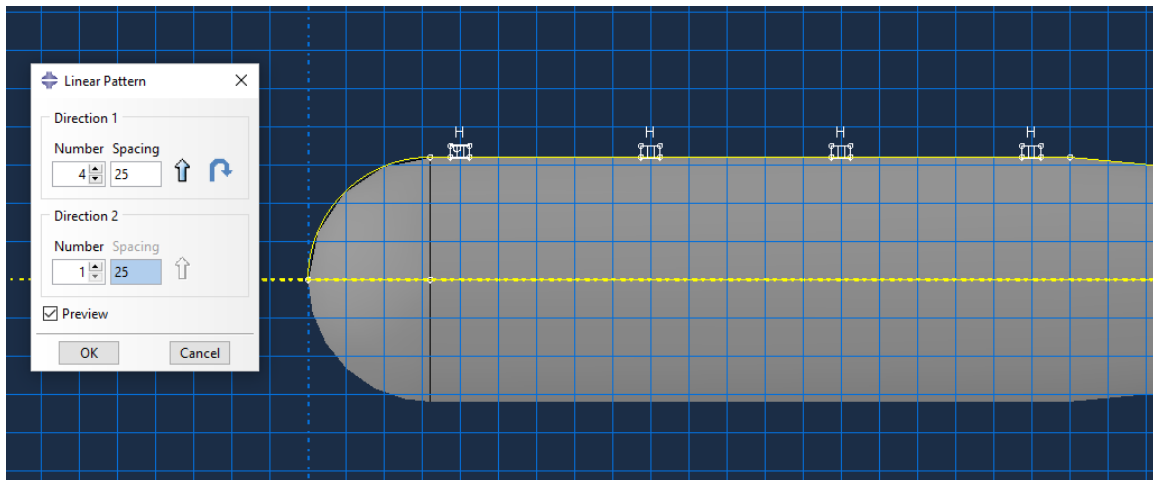


Figure 21 Répétition

Faire les mêmes étapes que le premier renfort

Résultat

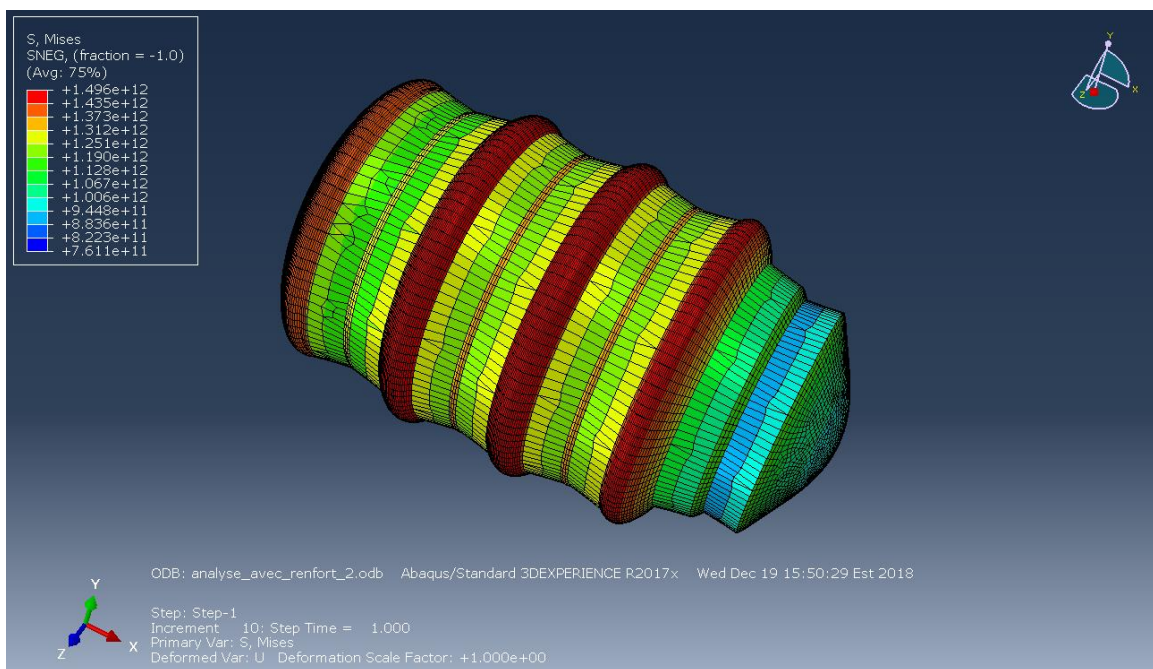


Figure 22 Contrainte

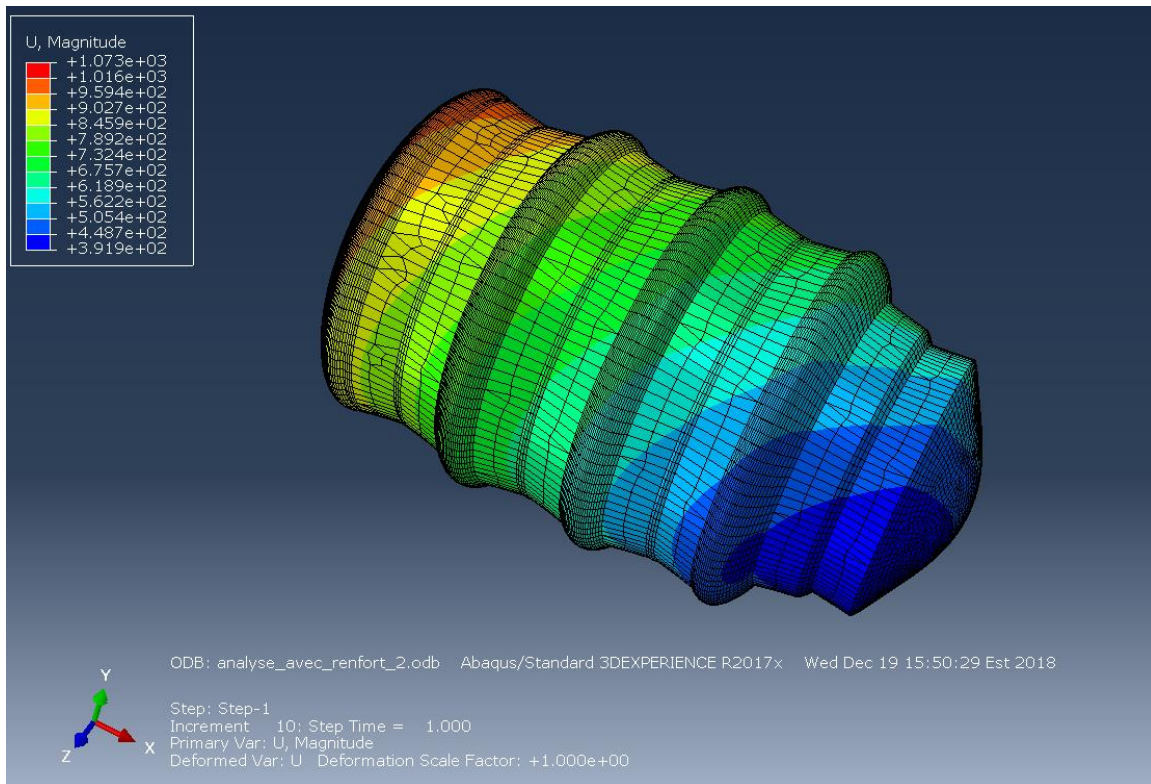


Figure 23 Déplacement

Troisième renfort

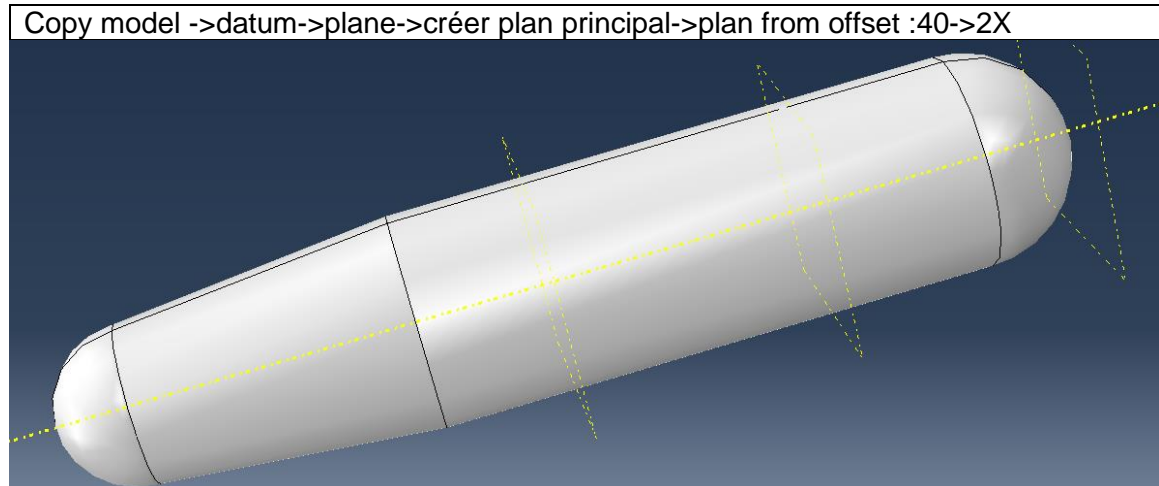


Figure 24 modèle

Shape ->solid->extrude->Skeetch ->cercle même diamètre que la coque -> depth :1
sur les deux plans->Shape->from solid->faire sur l'autre plan

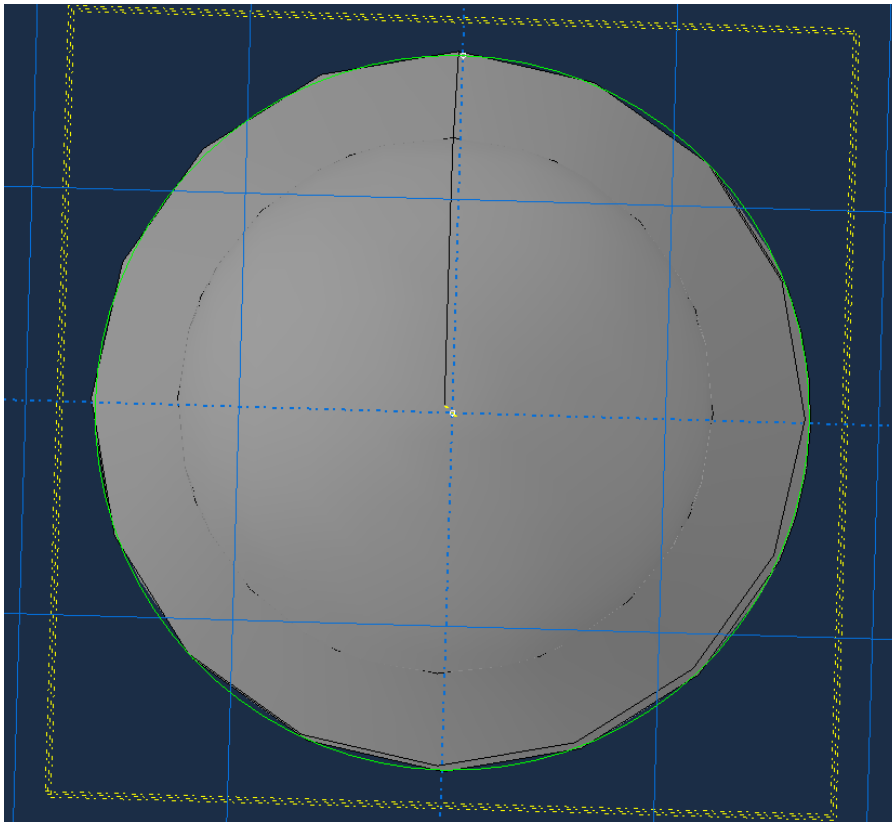


Figure 25 sketch renfort

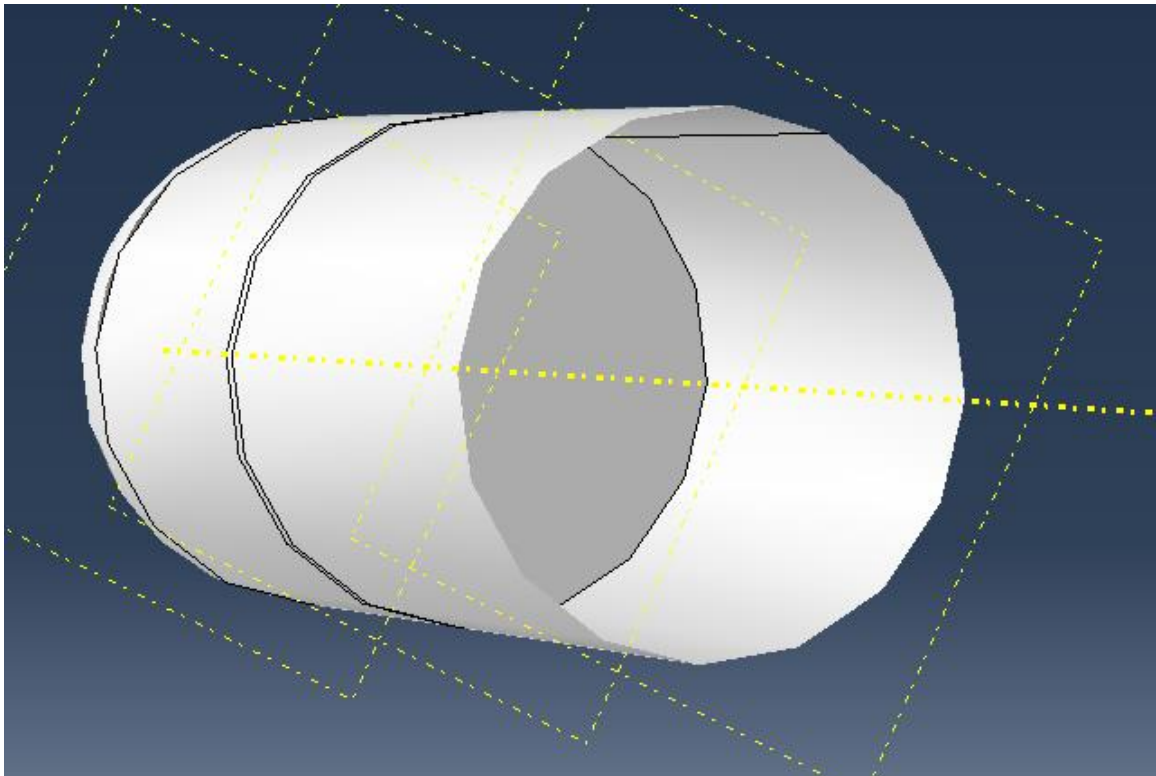


Figure 26 coupe afin de voir le renfort

Faire les même étape que les autres renforts

Résultat

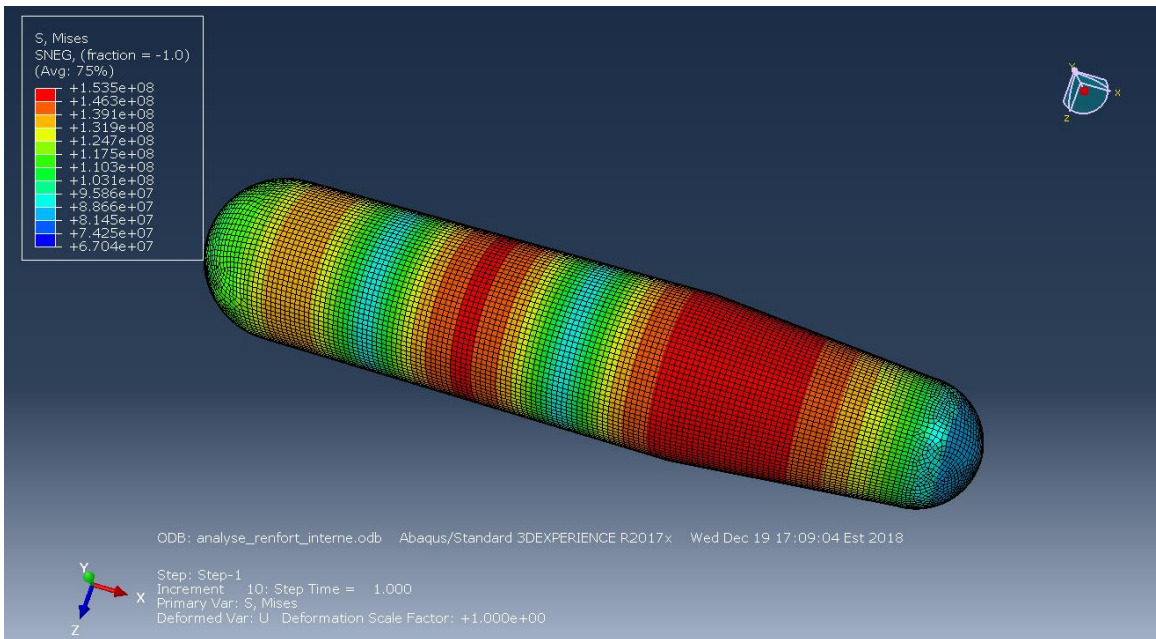


Figure 27 Contrainte

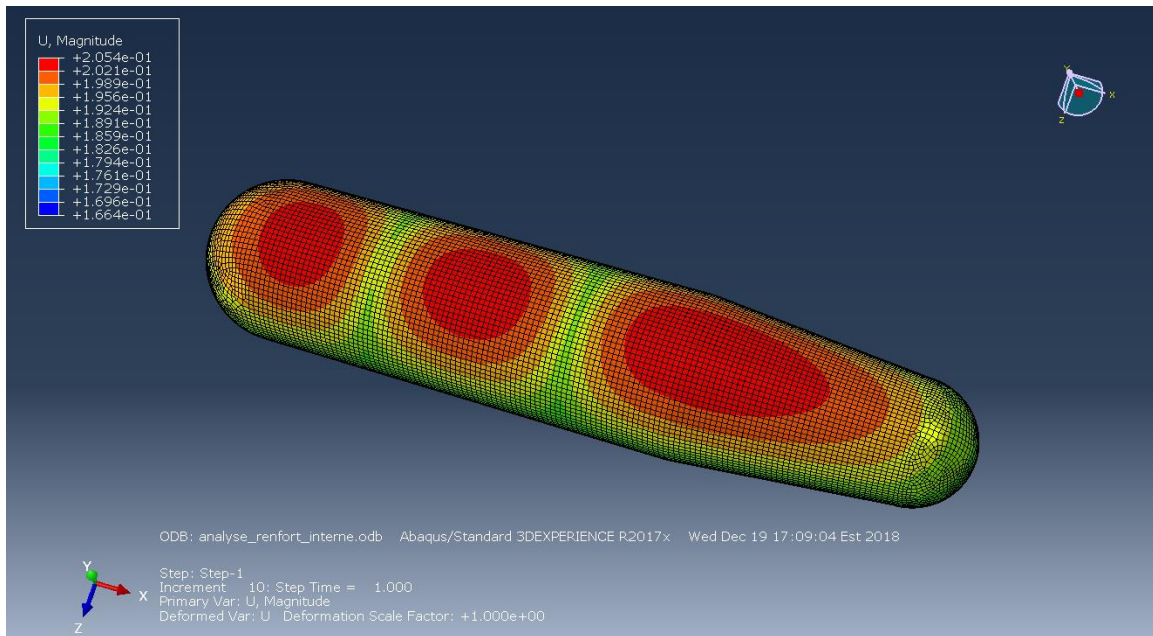


Figure 28 Déplacement

Discussion

Une géométrie lisse est essentielle afin de maintenir l'intégrité d'une coque de pression, car la contrainte augmente d'un facteur 10^5 avec une forme non lisse. Par contre mettre des renfort interne réduit un peu la contrainte (facteur environ 0.8), par contre l'ajout de renfort devrait probablement réduire encore la contrainte.

Optimisation

Optimisation -> optimisation task -> sizing

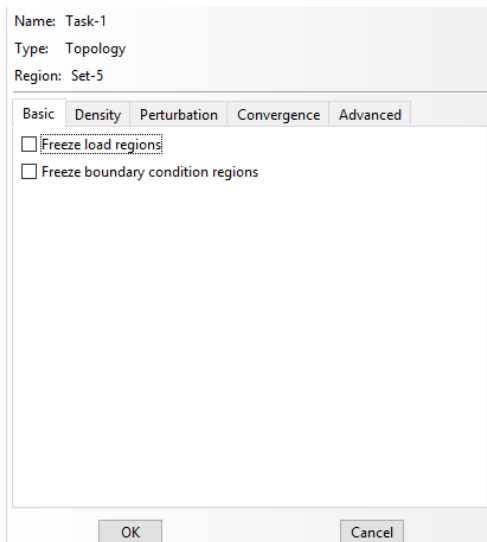


Figure 29 type d'optimisation

Design response -> create->whole model->weight, Strain energy

✓	D-Response-1	Single-term	Sum	Weight
✓	D-Response-2	Single-term	Sum	Strain energy

Figure 30 Réponse

Objective Function ->ok

Name: Objective-1
Task: Task-1 (Sizing, General)
Target: Minimize design response values
Design Response

	Name	Weight	Reference Value	Type
✓	D-Response-2	1	0	Strain energy

Figure 31 Objectif

Optimisation constraint -> 0.80 fraction

Name: Opt-Constraint-1
Task: Task-1 (Sizing, General)
Design Response
Name: D-Response-2
Type: Strain energy

Constrain the response to:

☐ A value \leq :

☒ A fraction of the initial value \leq :

☐ A value \geq :

☐ A fraction of the initial value \geq :

Figure 32 contrainte d'analyse

Optimisation->geometric restriction

Name: Restrict-1
Type: Thickness control (Sizing)
Task: Task-1 (Sizing, General)
Region: (Picked)

Restriction:

☐ Thickness by value
☒ Thickness by fraction

☐ Minimum: 1
☒ Maximum: 2

Figure 33 Restriction

Job->optimisation process manager

Name: Opt-Process-1
Model: optimisation_renfort_interne
Task: Task-1 (Topology, General)
Description:

Optimization Submission Memory Parallelization

Controls

Maximum cycles: 10

Data save: ☒ Specify cycles
☒ Initial (design cycle 0) ☐ First ☒ Last ☐ Every cycle
☐ Every cycle

Figure 34 optimisation

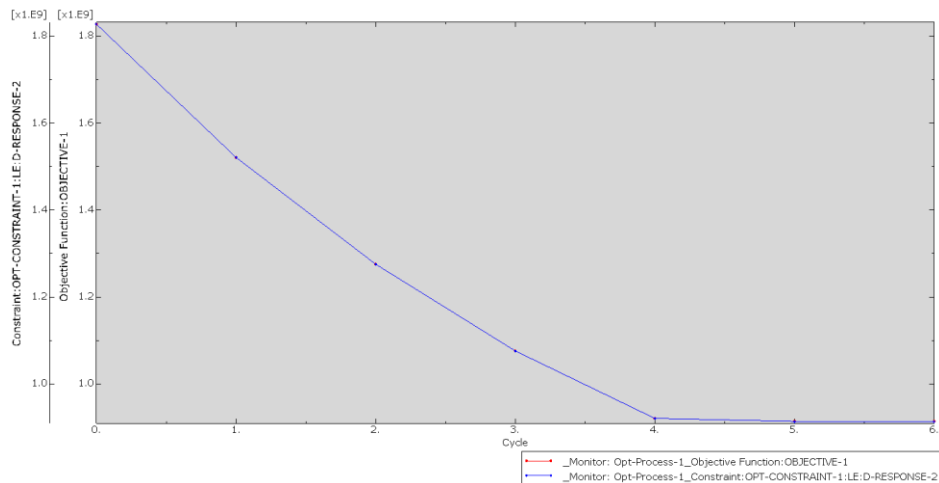


Figure 35 Graphique de convergence

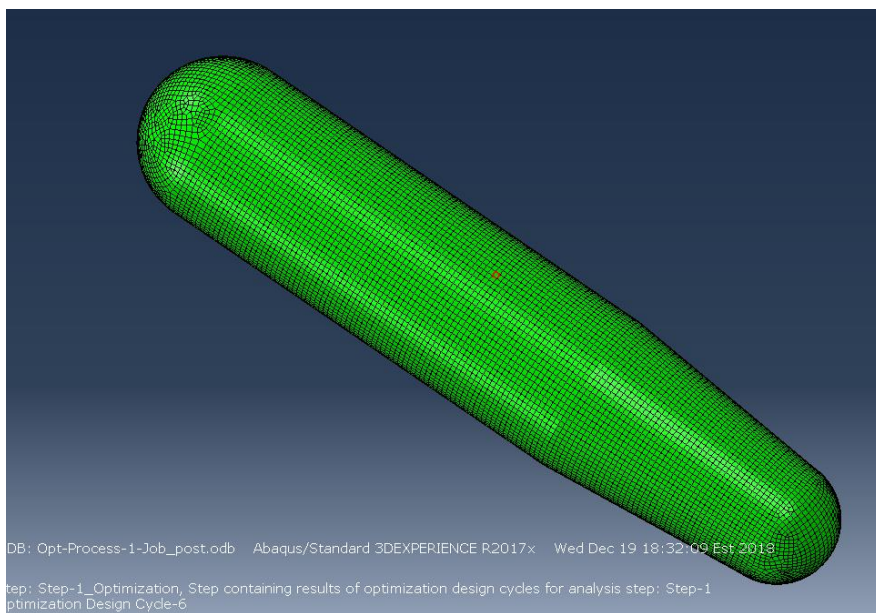


Figure 36 modèle optimisé

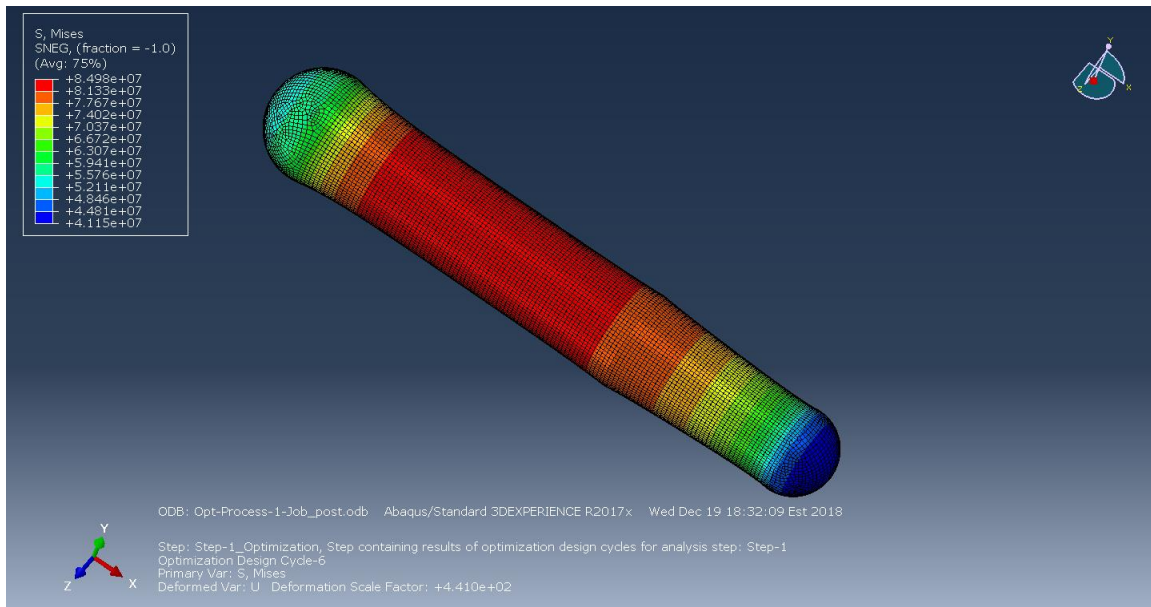


Figure 37 Contrainte

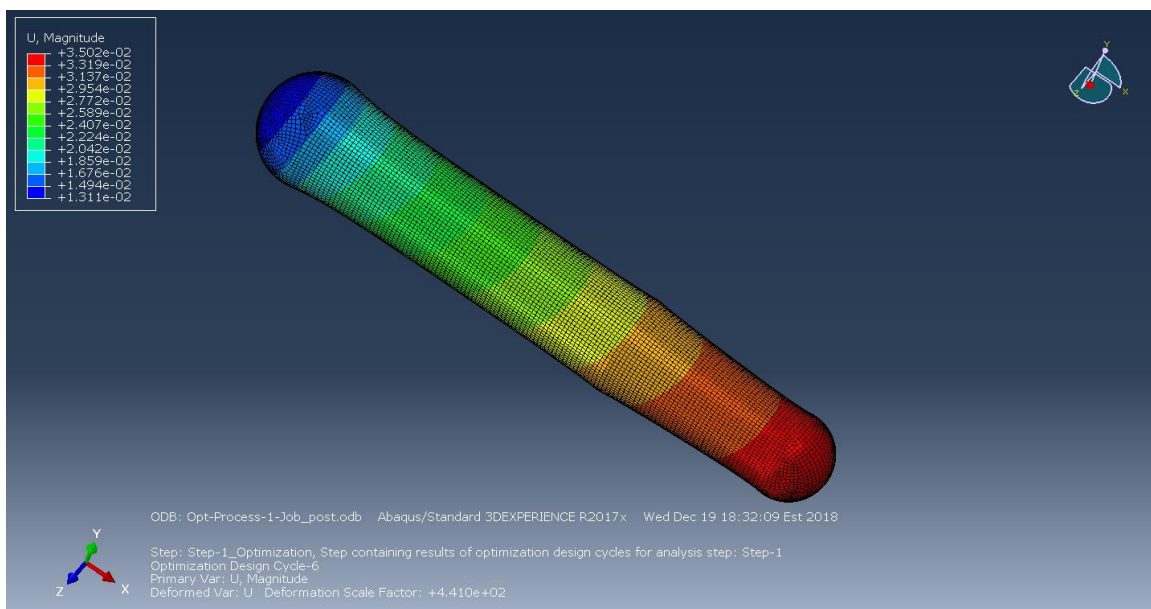


Figure 38 Déplacement

Discussion

L'optimisation a permis de réduire la contrainte d'un facteur d'environ 1.5.

Discussion et apport personnel

D'après les modélisations un modèle lisse avec des renforts internes en titane est préférable. Le modèle analytique a permis de montrer que l'analyse Abaqus obtient le bon ordre de grandeur. Ce travail de synthèse m'as permis d'apprendre à programmer un script python sur Abaqus ainsi que de consolider mes connaissances sur Abaqus. Par contre, plus de connaissance sur les coques de pression d'un sous-marin aurait très

intéressant afin d'en modéliser une plus cohérente. Une modélisation d'une coque légère intégrer avec la coque de pression aurait aussi été très intéressant.

Référence

Matweb (2018). <http://asm.matweb.com/search/SpecificMaterial.asp?bassnum=mtp641>

Matweb

(2018). <http://asm.matweb.com/search/SpecificMaterial.asp?bassnum=ma6061t6>

Matweb (2018). <http://www.matweb.com/search/datasheet.aspx?bassnum=MS0001>

Wikipédia (2017). https://en.wikipedia.org/wiki/Submarine_depth_ratings.

Arentzen.

Scribd (2018). <https://fr.scribd.com/document/337311688/Naval-Arch-Aspect-of-Submarine-Design>.

John R. MacKay(2007) .Structural Analysis and Design of PressureHulls: the State of the Art and Future Trends