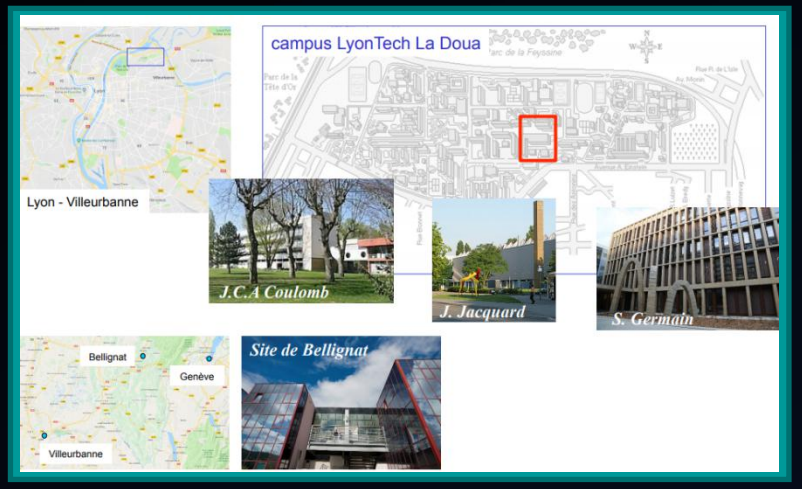
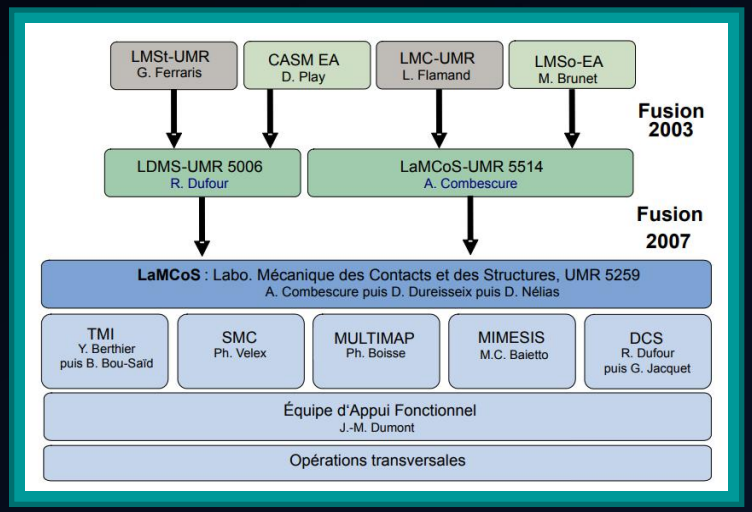
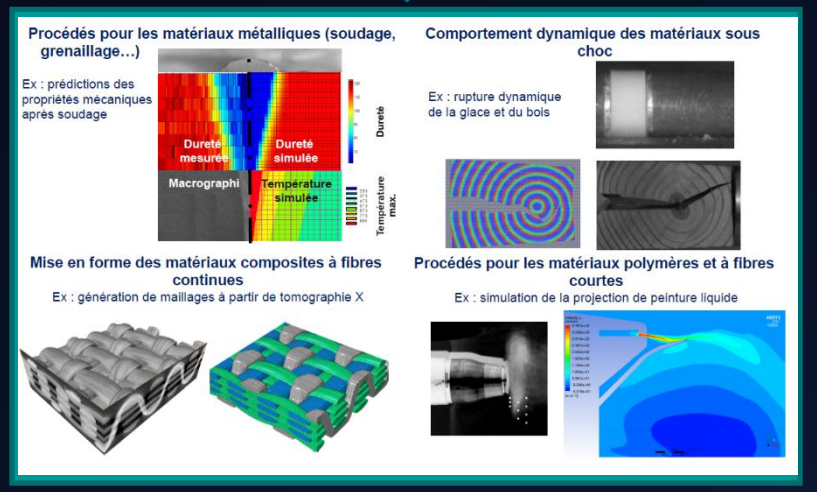
Calculs multi-échelles pour des composites à renfort interlock sur la base FFT et FEM

Soutenance de stage de fin d'étude au LaMCoS

Zhang Xunjie
Encadré par: M.Naim Naouar, M.Nawfal Blal
Tuteur pédagogique: M.Lionel Frossard







02 Avr - 29 Sep
M.Naïm Naouar
M.Nawfal Blal
M.Lionel Frossard
Marion, Soukaina, C.shuai

Avril-Mai :

Etude bibliographique
Installation d'équipement

Juin-Juillet :

Méthode FFT

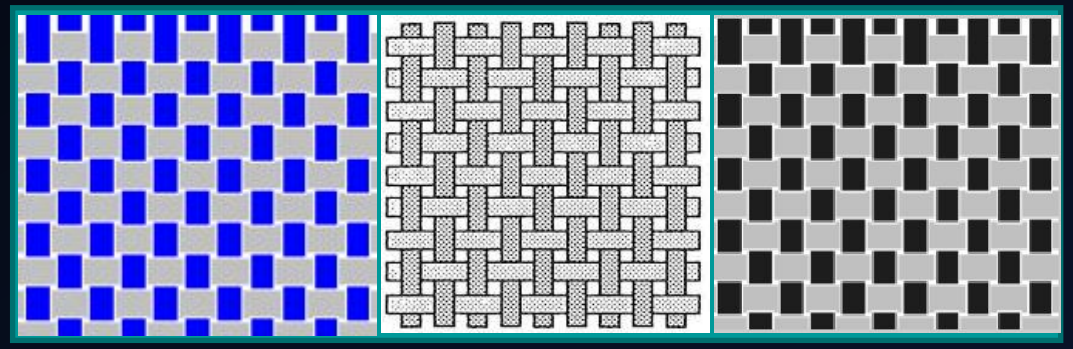
Août :

Méthode FEM

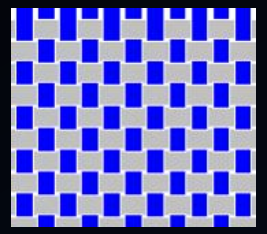
Septembre :

Rapport
Présentation

- Etude bibliographique
- Implementation de la solution d'un problème non-linéaire par la FFT
- Comparaison avec la solution MEF
- Développement d'une solution multi-échelle hybride MEF/FFT



FFT : Méthode



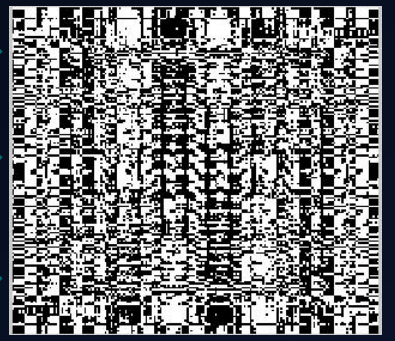
$$\widehat{f}(\xi) = \sum_{x=0}^{N-1} f(x) e^{-i2\pi\frac{\xi x}{N}}$$

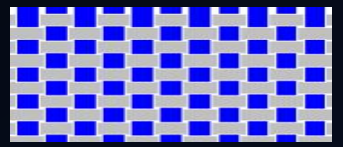
$$\widehat{f}(\xi_1,\xi_2) = \sum_{x_1=0}^{N_1-1}\sum_{x_2=0}^{N_2-1} f(x_1,x_2) e^{-i2\pi(\frac{\xi_1 x_1}{N_1}+\frac{\xi_2 x_2}{N_2})}$$

Traitement

Boucle

Vérifier



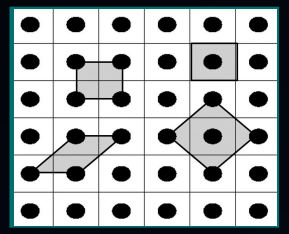


C^{hom}

$$f(x) = \frac{1}{N}\sum_{\xi=0}^{N-1} \widehat{f}(\xi) e^{i2\pi\frac{\xi x}{N}}$$

$$f(x_1,x_2) = \frac{1}{N}\sum_{\xi_1=0}^{N_1-1}\sum_{\xi_2=0}^{N_2-1} \widehat{f}(\xi_1,\xi_2) e^{i2\pi(\frac{\xi_1 x_1}{N_1}+\frac{\xi_2 x_2}{N_2})}$$

FFT : Simulation



```
% ---------MAIN----------------------------------------------------------
tic ;
clc ;
close all ;
clear all ;
...
...
...

while 1
    Sigma_i = FFT2(sigma_i) ;
    Epsilon_i = FFT2(epsilon_i);
    erreur = convergence_test(X,Y,Sigma_i)
    if erreur < 1e-4
        break ;
    end
    Epsilon_i1 = Epsilon_i - produ2(X,Y,Sigma_i,Epsilon_i,E,lambda0,mu0) ;
    epsilon_i1 = iFFT2(Epsilon_i1) ;
    sigma_i1 = produ1(epsilon_i1,C1,C2,im) ;
    epsilon_i = epsilon_i1 ;
    sigma_i = sigma_i1 ;
    ite=ite+1
    ee = [ee erreur] ;
end

...
...
```

Initialisation

$$\varepsilon^0(x_d) = E$$
$$\sigma^0(x_d) = c(x_d) : \varepsilon^0(x_d)$$

Itération

ε^i et σ^i sont connus pour tous les pixels

$\widehat{\sigma}^i = FFF(\sigma^i)$

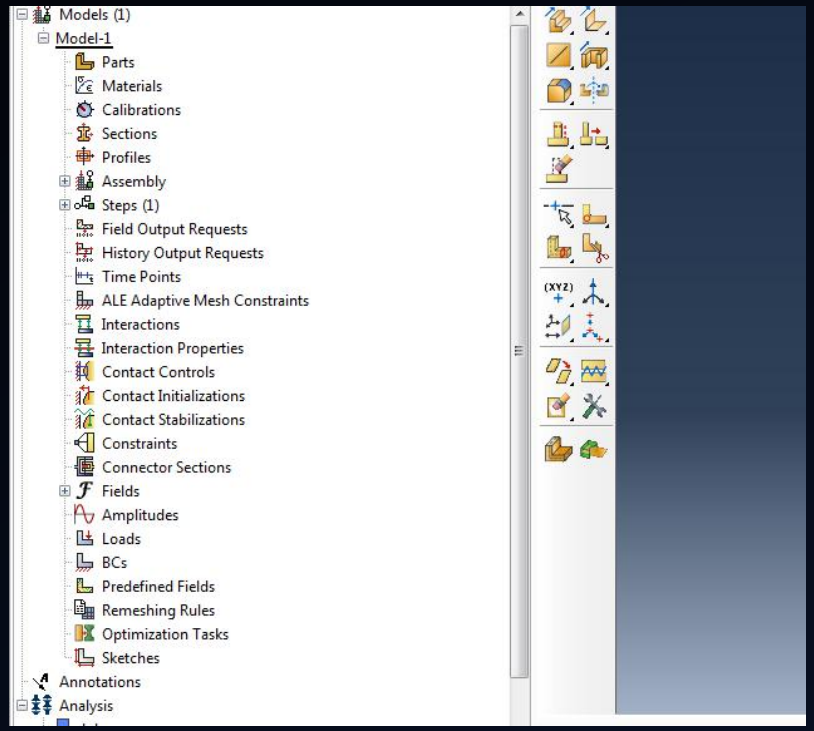
convergence test

$$\widehat{\varepsilon}^{i+1}(\xi_d) = \widehat{\varepsilon}^i(\xi_d) - \widehat{\Gamma}^0(\xi_d) : \widehat{\sigma}^i(\xi_d) \qquad \forall \xi_d \neq 0$$
$$\widehat{\varepsilon}^{i+1}(0) = E$$
$$\varepsilon^{i+1} = iFFF(\widehat{\varepsilon}^{i+1})$$
$$\sigma^{i+1}(x_d) = c(x_d) : \widehat{\varepsilon}^{i+1}(x_d) \qquad \forall x_d \in V$$

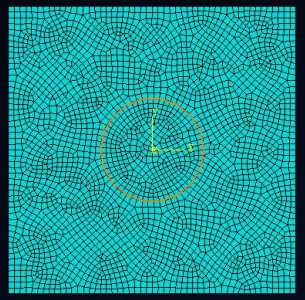
```
function y = C(lambda,mu)

function y = convergence_test(X,Y,Sigma_i)

function SOLU = FFF(A,X,Y,x,y)

function SOLU = iFFF(A,X,Y,x,y)

function make_pixel(N1,N2,vol)

function y = produ1(epsilon_i,C1,C2,im)

function y = produ2(X,Y,Sigma_i,Epsilon_i,E,lambda0,mu0)

function y = xi(N,T)
```

FEM : Abaqus

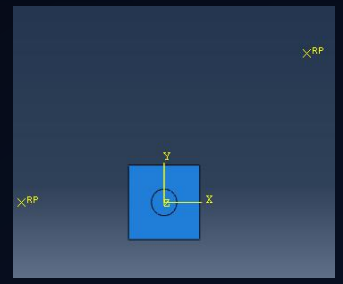
Interface



Maillage



Point référence



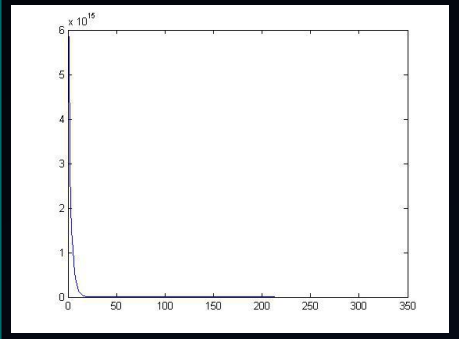
*Equation
N
$P, i, A_1, Q, j, A_2, \cdots, R, k, A_N$

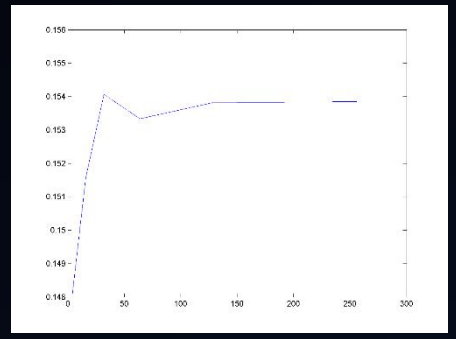
job2308.inp

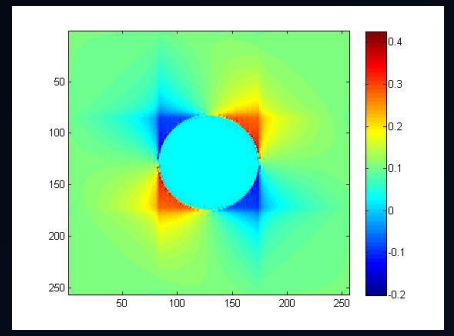
volume.rpt
deformation.rpt

```
@author: xzhang
"""
import numpy as np
f = open("C:/Temp/strain.rpt",'r')
for i in range(19):
    a = f.readline()
E11 = []
a = f.readline().split()
...
...
...
...
```

FFT :

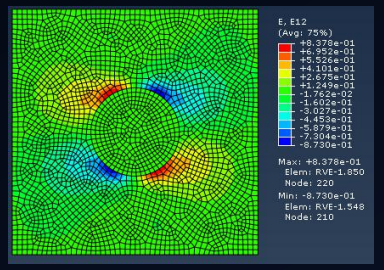


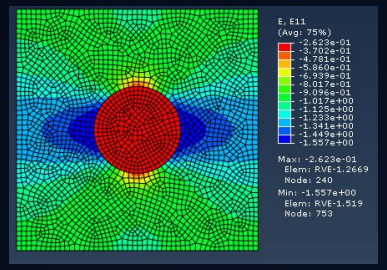


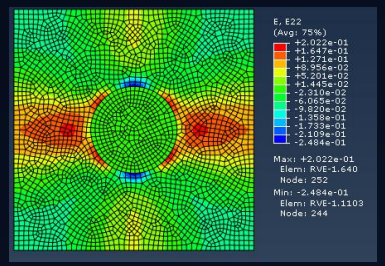


$$C^{hom} = 1.0e^{10} * \begin{bmatrix} 2.5583 & -2.5583 & 2.5583 \\ -2.5583 & 2.5583 & -2.5583 \\ 2.5583 & -2.5583 & 2.5583 \end{bmatrix}$$

FEM :








```
In [11]:
runfile('C:/Users/xzhang/Stage_LAMCOS/Abaqus_methode1/def_vol
ume.py', wdir='C:/Users/xzhang/Stage_LAMCOS/Abaqus_methode1')
-2.31941207406e-07 2.52567298642e-11 -2.31941207406e-07
```

- Missions
- Résultats
- Avenir

Merci pour votre attention !

Question ?