

Rapport TP X01

TP4

Aurélien Valade

1 Introduction

Dans ce rapport, nous allons nous pencher sur la description et l'implémentation d'un code de résolutions d'équations aux dérivées partielles par méthode d'éléments finis et résolution hétérogène multi-échelles, termes que nous détaillerons plus loin. Ce travail se présente en deux parties : tout d'abord en une analyse du papier de [?], puis une tentative de ré-implémentation de certaines parties de ce code à l'aide des routines développées dans les TP précédents. Contrairement au papier qui présente des méthodes pour différents types de maillage en deux ou trois dimensions, nous nous bornerons à des explications en deux dimensions avec des maillages de simplexes, c'est à dire triangulaires dans notre cas.

2 Analyse du papier & méthodologie du code

Ce papier a pour but l'implémentation d'un code MATLAB simple et multifonctions à mettre à disposition du public. Celui ci permettrait la résolution d'EDP dont la physique amène à considérer des phénomènes physiques couplés sur plusieurs échelles d'espaces distinctes. Pour la résolution de tels problèmes, une approche frontale serait donc de discrétiser tout l'espace avec un maillage dont les cellules seraient à l'échelle du phénomène physique le plus petit, mais cela amènerait à d'énormes problèmes numériques très coûteux à résoudre. Parfois la physique demande même à ce que la taille du phénomène microscopique tende vers zéro, il est alors impossible d'attaquer le problème brutalement et il faut donc créer un meilleur cadre analytique et trouver des méthodes numériques plus efficaces quitte à faire des approximations.

La littérature semble être fournie en ce qui concerne l'approche analytique de ce genre de problème multi-échelles, cependant peu d'outils numériques seraient disponibles aujourd'hui, ce à quoi le travail présenté dans ce papier veut remédier en proposant ce code MATLAB court, modulable et parallélisable.

2.1 Cadre mathématique de la résolution

2.1.1 Présentation de l'équation

L'étude présentée dans ce papier porte sur des méthodes d'éléments finis (FE) pour la résolution de problèmes elliptiques et paraboliques comme suit.

Problème 1. Soient les sous espaces $\Omega \subset \mathbb{R}^d$ et $Y = [0, 1]^d$. Soient $\varepsilon > 0$, $\alpha \in \{0, 1\}$ et soient les fonctions sur ces espaces $f : \Omega \rightarrow \mathbb{R}$, $A^\varepsilon : \Omega \times Y \rightarrow \mathcal{M}_d(\mathbb{R})$ telle que $A^\varepsilon = (A^\varepsilon)^T$ et $g_{N,D} : \partial\Omega_{N,D} \rightarrow \mathbb{R}$.

Trouver u tel que

$$\begin{cases} \alpha \partial_t u^\varepsilon - \nabla(A^\varepsilon \nabla u^\varepsilon) = f & \text{sur } \Omega \\ u^\varepsilon = g_D & \text{sur } \partial\Omega_D \\ n \cdot (A^\varepsilon \nabla u^\varepsilon) = g_N & \text{sur } \partial\Omega_N \end{cases}$$

Remarque 1. Il s'agit d'un problème elliptique si $\alpha = 0$, sinon il s'agit d'un problème parabolique.

Remarque 2. La dépendance en ε de A^ε n'est pas forcément de la forme vue en cours, i.e. périodique $A^\varepsilon(x/\varepsilon)$, mais elle peut être quasi-périodique $A^\varepsilon(x, x/\varepsilon)$, ou bien même implicite, bien qu'on ne considère dans la suite que ce cas simple. Il est cependant nécessaire d'avoir pour tout ε les propriétés suivantes :

$$\exists \lambda, \Lambda > 0 \quad \text{s.t.} \quad \lambda \xi^2 < (A^\varepsilon \xi) \cdot \xi < \Lambda \xi^2 \quad \forall \xi \in \mathbb{R}^d$$

en tous points pour pouvoir prouver que le problème variationnel vu en [subsubsection 2.1.2](#) est bien posé.

Remarque 3. Dans notre code on traitera uniquement le cas où $\partial\Omega_N = \emptyset$.

Cette formulation permet de mettre en exergue la dépendance en ε qui est l'échelle la plus petite du problème, et qui tend usuellement vers zéro. C'est cette dépendance qui est au centre du questionnement. En effet il ne suffit pas de trouver une solution u^ε explicite en ε et de faire tendre ε vers zéro pour trouver u^0 – ce qui est de plus impossible analytiquement dans le cas général. La convergence n'est généralement que faible dans L^2 :

$$\lim_{\varepsilon \rightarrow 0} \int_{\Omega} u^\varepsilon \phi = \int_{\Omega} u^0 \phi \quad \forall \phi \in L^2(\Omega).$$

Les deux méthodes présentées plus loin permettent de palier à ce problème par des méthodes analytiques amenant à des implémentations numériques modifiées.

2.1.2 Problème variationnel associé

On va poser ici la formulation variationnelle discrète du problème pour mieux voir les éléments qui nous intéressent réellement, à savoir les coefficients de la matrice de rigidité.

Par une démonstration vue plusieurs fois en cours dans un cadre plus restreint¹, on obtient que le problème 1 est équivalent à

$$\underbrace{\int_{\Omega} (A^\varepsilon \nabla u^\varepsilon) \cdot \nabla \phi}_{B^\varepsilon(u^\varepsilon, \phi)} = \underbrace{\int_{\Omega} f \phi + \int_{\partial\Omega_N} g_N \phi - \int_{\Omega} (A^\varepsilon \nabla g_D) \cdot \nabla \phi}_{l^\varepsilon(\phi)} \quad \forall \phi \in L^2(\Omega). \quad (1)$$

Remarque 4. On peut réécrire immédiatement

$$l^\varepsilon(\phi) = l(\phi) - B^\varepsilon(g_D, \phi).$$

Sous certaines conditions de coercitivité et de continuité des fonctions (bi)linéaires en présence, on peut montrer que le problème est bien posé à l'aide du théorème de Lax-Milgram.

Il apparaît que le point centrale de la difficulté de cette équation est dans la fonction bilinéaire B^ε . Ce sont en effet ces coefficients que l'on cherchera à calculer par la méthode la plus adaptée pour quand ε tend vers 0.

Avant cela, discrétisons cette équation sur une base finie de fonctions de $L^2(\Omega)$. On se donne un maillage de simplexe en deux dimensions². On note les sommets de ce maillage les $\{S_i\}_{1,N}$ et \mathcal{T}_h l'ensemble des simplexe. On définit l'ensemble des sommets les fonctions de $\mathbb{R}^g[X]$ avec $g \in \mathbb{N}$ le degré des polynômes choisis. Ces fonctions, notées $\{\varphi_i\}_{1,N}$ par la suite sont telles que

$$\varphi_i(S_j) = \delta_{i,j} \quad \forall i, j \in [1, N]^2.$$

On projette u^ε sur cette bases et on utilise les $\{\varphi_i\}_{1,N}$ comme ϕ pour obtenir le système linéaire

$$\mathbb{K}^\varepsilon U^\varepsilon = L^\varepsilon$$

avec $\mathbb{K}_{ij}^\varepsilon = B^\varepsilon(\varphi_i, \varphi_j)$, $U_i^\varepsilon = u^\varepsilon(S_i)$ et $L_i^\varepsilon = l^\varepsilon(\varphi_i)$.

¹Multiplication par ϕ , Green-Ostrogradski deux fois dont la deuxième pour changer l'intégrale sur g_D .

²Le papier susmentionné va plus loin en considérant d'autres types de maillages, en 2D et en 3D.

2.2 Méthode d'homogénéisation classique

Le concept développé en homogénéisation pure est de trouver une méthode pour calculer le tenseur A^{hom} en tout point pour pouvoir ensuite résoudre le problème dérivé qui nous donne la solution $u^{hom} = \lim_{\varepsilon \rightarrow 0} u^\varepsilon$:

Problème 2. *Trouver u^{hom} tel que*

$$\alpha \partial_t u^{hom} - \nabla(A^{hom} \nabla u^{hom}) = f$$

L'avantage est ici que ε a complètement disparu de l'équation et donc qu'un maillage adapté aux grandes échelles seules suffit. Cependant il faut pour cela calculer $A^{hom}(\bar{x})$ au préalable *i.e.* avant la résolution du macro problème. Pour cela on résout les problèmes dits de cellule en tout point $\bar{x} \in \Omega$ pour trouver les $\{w_i\}_{1,d}$:

$$\int_Y A^1(\bar{x}, y) \nabla w_i(\bar{x}, y) \nabla \phi(y) dy = \int_Y A^1(\bar{x}, y) e_i \nabla \phi(y) dy \quad \forall \phi \in L^2(\Omega) \quad \forall i \in \{1, d\} \quad (2)$$

puis on calcule le tenseur homogénéisé

$$A_{jk}^{hom}(\bar{x}) = \int_Y A(\bar{x}, y) (e_k + \nabla_y w_k(\bar{x}, y)) (e_j + \nabla_y w_j(\bar{x}, y)) dy \quad (3)$$

et on peut finalement résoudre le système linéaire homogénéisé :

$$\mathbb{K}U = L$$

avec $\mathbb{K}_{ij} = \int_\Omega (A^{hom} \varphi_i) \cdot \varphi_j$, $U_i = u^{hom}(S_i)$ et $L_i = l(\varphi_i) - \int_\Omega (A^{hom} g_D) \cdot \varphi_i$.

Remarque 5. *Dans notre cas, $d = 2$.*

Le gros inconvénient de cette méthode étant qu'il faut préalablement calculer le tenseur homogénéisé A^{hom} théoriquement en tout point de Ω de l'espace discrétisé.

2.3 Différences et apport de la nouvelle méthode

La méthode de la FE-HMM (*finite elements heterogenous multi-scale method*) présente tous les avantages d'une méthode d'homogénéisation, mais permet en plus d'éviter le calcul de $A^{hom}(\bar{x})$ au préalable. On peut ainsi travailler exclusivement avec comme entrée les données initiales, et calculer le moins de données intermédiaires. La méthode n'est pas mathématiquement plus légère mais permet de faire en moins d'opérations découplées le même travail. Ici, on prévoit de calculer directement la contribution de chaque cellule à la matrice de rigidité globale. De plus ce calcul est complètement parallélisable !

Deux différentes échelles se distinguent :

- la macro-échelle, de taille caractéristique de maillage H
- et la micro-échelle, de taille caractéristique de maillage h ,

la première étant celle associée du système physique, et la seconde étant une échelle arbitrairement plus petite dont l'usage sera précisé plus loin.

Remarque 6. *On parlera dans la suite de macro-maillage (resp. micro-) composé de macro-cellules (resp. micro-) par raccourci de langage pour désigner les quantités liées à la macro-échelle (resp. micro-).*

On aurait a priori envie d'associer directement la micro-échelle à la taille d'une maille de la macro, *i.e.* de mailler chaque macro-cellule par un micro-maillage. Cependant la méthode proposée est un peu différente. On va choisir dans macro-cellule un ensemble de points pour faire une quadrature. Sont proposées deux quadratures :

- une quadrature avec un unique point au centre de la macro cellule
- une quadrature de Legendre à deux points en chaque direction de l'espace, on aura donc 4 points en $\{(1/2 \pm \sqrt{3}/6, 1/2 \pm \sqrt{3}/6)\}$ en considérant une maille centrée en $(0,0)$,

auxquelles on pourrait ajouter d'autres quadratures d'ordre plus élevé par exemple. On va considérer dans la suite que l'on travaille dans le cas de la quadrature à quarts points.

Remarque 7. *Nous allons dans notre code reprendre la quadrature à quatre points proposée dans le TP1 à la place de la quadrature de Gauss-Legendre.*

Autour de chaque point de cette quadrature on forme un carré de côté $\delta \ll H$. C'est sur ces espaces que l'on va résoudre les problèmes de cellules, et c'est cette échelle que nous appelons micro-échelle.

Remarque 8. *On utilise dans la suite des micro-maillages de côté $\delta = \varepsilon$, ce qui amène d'après le papier à un code robuste et indépendant de ε .*

2.4 Modification analytique des problèmes de cellule

Les problèmes de cellule sur lesquels on travaille sont des problèmes modifiés qui permettent, comme précisé précédemment, de ne pas calculer A^{hom} mais directement la contribution locale à la matrice de rigidité à partir des données initiales. Les calculs suivant démontrent comment, en partant des équations vues en cours, on peut retrouver les problèmes de cellule modifiés qui seront résolus dans le code dans un cadre plus général puisqu'on se situe dans le cadre où A^ε est quasi-périodique. Ces problèmes sont au nombre de sommets par cellule. Comme on utilise uniquement des simplexes de cimentations deux, on aura donc trois problèmes modifiés par macro-cellule. Pour faire l'analogie avec les codes vus dans les TP précédents, ces problèmes permettent en réalité de conduire directement les matrices élémentaires $\mathbf{Ke1}$ de la matrice de rigidité.

On veut montrer d'abord que l'Equation 3 peut se réécrire sous la forme de l'Equation 4:

$$A_{jk}^{hom}(\bar{x}) = \frac{1}{|Y_\varepsilon|} \int_{Y_\varepsilon} A(\bar{x}, x/\varepsilon) (e_k + \nabla_x \eta_k(\bar{x}, x/\varepsilon)) (e_j + \nabla_x \eta_j(\bar{x}, x/\varepsilon)) dx \quad (4)$$

avec

$$\eta_i = \varepsilon w_i \quad \forall i \in \{1, 2\}$$

On procède par changement de variable : $y \rightarrow x/\varepsilon$ et en faisant les transformations suivantes

- $dy \rightarrow dx/\varepsilon = dx/|Y_\varepsilon|$,
- $y = (-1/2, -1/2) \rightarrow x = (-\varepsilon/2, -\varepsilon/2) \implies Y \rightarrow Y_\varepsilon = (-\varepsilon/2, -\varepsilon/2)^2$,
- $\nabla_y = \frac{\partial}{\partial y} = \frac{\partial x}{\partial y} \frac{\partial}{\partial x} = \varepsilon \nabla_x$,

la réécriture est immédiate.

On veut ensuite montrer que le problème de cellule est "translatable", c'est à dire que l'on peut travailler autour d'un point quelconque \bar{x} . Cela est direct si on remarque que, la fonction étant périodique et l'intervalle d'intégrations étant précisément un intervalle de périodicité de la fonction, la translation de ce dernier n'influe pas sur l'intégrale :

$$\int_{Y_\varepsilon(\bar{x})} A(\bar{x}, x/\varepsilon) \nabla_x \eta_i(\bar{x}, x/\varepsilon) \nabla_x \phi(x) dx = - \int_{Y_\varepsilon(\bar{x})} A(\bar{x}, x/\varepsilon) e_i \nabla_x \phi(x) dx \quad (5)$$

On retrouve alors l'équation sur Y avec les w_i des problèmes de cellule :

$$\int_Y A(y) \nabla_y w_i(y) \nabla_y \phi(\bar{x} + \varepsilon y) dy = - \int_Y A(y) e_i \nabla_y \phi(\bar{x} + \varepsilon y) dy \quad \forall i \in \{1, 2\}, \forall \phi \in V_\varepsilon(Y_\varepsilon)$$

mais on voudrait montrer cette égalité $\forall \tilde{\phi} \in V$, ce qui est équivalent car

$$\forall \tilde{\phi} \in V, \exists \phi \in V_\varepsilon(Y_\varepsilon) \quad \text{tq} \quad \tilde{\phi}(x) = \phi(\bar{x} + \varepsilon y) \text{ en tout points associés } x \rightarrow y$$

or l'Equation 5 étant vraie $\forall \phi \in V_\varepsilon(Y_\varepsilon)$, on peut bien écrire

$$\int_Y A(y) \nabla_y w_i(y) \nabla_y \phi(y) dy = - \int_Y A(y) e_i \nabla_y \phi(y) dy \quad \forall i \in \{1, 2\}, \forall \phi \in V. \quad (6)$$

Le but de la méthode étant d'éviter le calcul intermédiaire de A^{hom} , on va écrire analytiquement les coefficients de \mathbb{K} qui nous intéressent. Ainsi, es prochaines équations construisent la contribution d'un simplexe à un coefficients \mathbb{K}_{IJ} représentant le recouvrement entre deux éléments de la base globale appartenant noté dans la base locale w_i et w_j , $i, j \in [1, 3]$. Pour alléger le calcul, on pose les conventions d'écriture suivantes

- $A^{hom} = \begin{pmatrix} A_{11}^{hom} & A_{12}^{hom} \\ A_{21}^{hom} & A_{22}^{hom} \end{pmatrix}$
- $\partial_{x_i} \rightarrow \partial_i, \quad \forall i \in \{1, 2\}$
- $B = \begin{pmatrix} 1 + \partial_1 \eta_1 & \partial_1 \eta_2 \\ \partial_2 \eta_1 & 1 + \partial_2 \eta_2 \end{pmatrix} = (e_1 + \nabla_x \eta_1 \quad e_2 + \nabla_x \eta_2) = (B_1 \quad B_2) = I + \nabla_x \eta$

et on ne note pas les arguments des fonctions. Commençons par écrire $A^{hom} \nabla_x w_j$:

$$\begin{aligned} A^{hom} \nabla_x w_j &= \begin{pmatrix} A_{11}^{hom} & A_{12}^{hom} \\ A_{21}^{hom} & A_{22}^{hom} \end{pmatrix} \begin{pmatrix} \partial_1 w_j \\ \partial_2 w_j \end{pmatrix} \\ &= \begin{pmatrix} A_{11}^{hom} \partial_1 w_i + A_{12}^{hom} \partial_2 w_i \\ A_{21}^{hom} \partial_1 w_i + A_{22}^{hom} \partial_2 w_i \end{pmatrix} \end{aligned} \quad (7)$$

on peut maintenant multiplier par la droite

$$\begin{aligned} A^{hom} \nabla_x w_j \nabla_x w_i &= (A_{11}^{hom} \partial_1 w_i + A_{12}^{hom} \partial_2 w_i) \partial_1 w_i + (A_{21}^{hom} \partial_1 w_i + A_{22}^{hom} \partial_2 w_i) \partial_2 w_i \\ &= \frac{1}{|Y_\varepsilon|} \int_{Y_\varepsilon} A(e_1 + \nabla_x \eta_1) \cdot (e_1 + \nabla_x \eta_1) \partial_1 w_j \partial_1 w_i + \\ &\quad A(e_2 + \nabla_x \eta_2) \cdot (e_1 + \nabla_x \eta_1) \partial_2 w_j \partial_1 w_i + \\ &\quad A(e_1 + \nabla_x \eta_1) \cdot (e_2 + \nabla_x \eta_2) \partial_1 w_j \partial_2 w_i + \\ &\quad A(e_2 + \nabla_x \eta_2) \cdot (e_2 + \nabla_x \eta_2) \partial_2 w_j \partial_2 w_i \\ &= \frac{1}{|Y_\varepsilon|} \int_{Y_\varepsilon} AB_1 \partial_1 w_j \cdot B_1 \partial_1 w_i + AB_2 \partial_2 w_j \cdot B_1 \partial_1 w_i + \\ &\quad AB_1 \partial_1 w_j \cdot B_2 \partial_2 w_i + AB_2 \partial_2 w_j \cdot B_2 \partial_2 w_i] \\ &= \frac{1}{|Y_\varepsilon|} \int_{Y_\varepsilon} A(B_1 \partial_1 w_j + B_2 \partial_2 w_j) \cdot B_1 \partial_1 w_i + \\ &\quad A(B_1 \partial_1 w_j + B_2 \partial_2 w_j) \cdot B_2 \partial_2 w_i + \\ &= \frac{1}{|Y_\varepsilon|} \int_{Y_\varepsilon} AB \nabla_x w_j \cdot B \nabla_x w_i \\ &= \frac{1}{|Y_\varepsilon|} \int_{Y_\varepsilon} A(I + \nabla_x \eta) \nabla_x w_j (I + \nabla_x \eta) \nabla_x w_i \end{aligned} \quad (8)$$

cette équation appliquée aux points corrects donne $\forall i, j \in [1, 3]$

$$\begin{aligned} A^{hom} \nabla_x w_j \nabla_x w_i &= \frac{1}{|Y_\varepsilon(x_{T,k})|} \int_{Y_\varepsilon(x_{T,k})} A(x_{T,k}, x/\varepsilon) (I + \nabla_x \eta(x_{T,k}, x/\varepsilon)) \nabla_x w_j(x_{T,k}) \\ &\quad (I + \nabla_x \eta(x_{T,k}, x/\varepsilon)) \nabla_x w_i(x_{T,k}) \end{aligned}$$

On veut enfin écrire le problème modifié dans la forme définitive où apparaît le couplage en fonctions de la micro-base et celles de la macro-base. En passant le membre de droite dans l'Equation 5, on retrouve

$$\begin{aligned} & \int_{Y_\varepsilon(x_{T,k})} A(x_{T,k}, x/\varepsilon) (e_i + \nabla_x \eta_i(x_{T,k}, x/\varepsilon)) \nabla_x \phi(x) dx = 0 \\ \iff & \int_{Y_\varepsilon(x_{T,k})} A(x_{T,k}, x/\varepsilon) B_i \nabla_x \phi(x) dx = 0 \end{aligned} \quad (9)$$

on peut donc écrire les couples d'équations scalaires $\forall i \in \{1, 2\}$, avec $l \in [1, 3]$.

$$\begin{aligned} & \begin{cases} \int_{Y_\varepsilon(x_{T,k})} A(x_{T,k}, x/\varepsilon) B_1(x_{T,k}, x/\varepsilon) \nabla_x \phi(x) (\nabla w_l)_1 dx = 0 & i = 1 \\ \int_{Y_\varepsilon(x_{T,k})} A(x_{T,k}, x/\varepsilon) B_2(x_{T,k}, x/\varepsilon) \nabla_x \phi(x) (\nabla w_l)_2 dx = 0 & i = 2 \end{cases} \\ \iff & \begin{cases} \int_{Y_\varepsilon(x_{T,k})} A(x_{T,k}, x/\varepsilon) B_1(x_{T,k}, x/\varepsilon) (\nabla w_l)_1 \nabla_x \phi(x) dx = 0 & i = 1 \\ \int_{Y_\varepsilon(x_{T,k})} A(x_{T,k}, x/\varepsilon) B_2(x_{T,k}, x/\varepsilon) (\nabla w_l)_2 \nabla_x \phi(x) dx = 0 & i = 2 \end{cases} \end{aligned} \quad (10)$$

d'où

$$\int_{Y_\varepsilon(x_{T,k})} A(x_{T,k}, x/\varepsilon) \nabla_x \tilde{w}_l(x_{T,k}, x/\varepsilon) \nabla_x \phi(x) dx = 0 \quad \forall l \in [1, 3] \quad (11)$$

avec

$$\nabla_x \tilde{w}_l(x_{T,k}, x/\varepsilon) = B(x_{T,k}, x/\varepsilon) \nabla_x w_l(x_{T,k}).$$

Il est important de noter qu'il ne s'agit plus de résoudre un problème elliptique comme jusqu'à présent, puisqu'une contrainte supplémentaire est imposée par les équations :

$$\tilde{w}_l \in w_{l,lin} + H_{\#}^1(Y_\varepsilon).$$

Remarque 9. Cette contrainte est importante, sans elle le problème serait mal posé, la fonction nulle étant une alternative à celle recherchée.

2.5 Résolution numérique des nouveaux problèmes de cellule

2.5.1 Système contraint

En suivant les recommandations du papier, on va donc changer le système à résoudre en passant par une méthode Lagrangienne. Cette nouvelle formulation devra donc prendre en compte les deux contraintes en présence :

- La pseudo-périodicité des problèmes de cellule modifiés
- La contrainte sur l'espace de définition de la solution.

Discretisons nos espaces, soient

- $\{\psi_i\}_{M_{mic}}$ la micro-base *i.e.* la base de $Y_\varepsilon(x_{T,k})$
- $\{\varphi_i\}_{1,2,3}$ la macro-base **locale**.

On projette sur la micro-base :

- $\tilde{w}_l = \sum \alpha_l^i \psi_i = \boldsymbol{\alpha}_l \cdot \boldsymbol{\psi}$ les solutions aux problèmes de cellule modifiés,
- $\varphi_l = \sum \beta_l^i \psi_i = \boldsymbol{\beta}_l \cdot \boldsymbol{\psi}$ les éléments de la base locale.

Les inconnues sont maintenant les vecteurs $\boldsymbol{\alpha}_l$, le reste étant connu ou aisément calculable.

Remarque 10. On rappelle qu'il y a donc trois vecteurs inconnus par points de quadrature, bien qu'on ne surcharge pas la notation avec un indice k sur les α_l .

On va poser la matrice \mathbb{D} qui regroupera nos contraintes, et on se donne la matrice de rigidité locale $\mathbb{K}^{T,k}$ calculée de manière classique sur l'intervalle $Y_\varepsilon(x_{T,k})$. On va chercher à résoudre les systèmes

$$\begin{cases} \mathbb{K}^{T,k} \alpha_l + \mathbb{D} \lambda_l = 0 \\ \mathbb{D}^t (\alpha_l - \beta_l) = 0 \end{cases} \quad \forall l \in [1, 3] \quad (12)$$

où $\lambda_l \in \mathbb{R}^C$ est un multiplicateur de Lagrange et C le nombre de contraintes. On va réécrire ces systèmes en un seul où les solutions sont concaténées

$$\mathbb{K}_c^{T,k} \alpha_c = \beta_c \quad (13)$$

avec

$$\mathbb{K}_c^{T,k} = \left(\begin{array}{c|c} \mathbb{K}^{T,k} & \mathbb{D} \\ \hline \mathbb{D}^t & 0 \end{array} \right), \quad \alpha_c = \begin{pmatrix} \alpha_1 & \alpha_2 & \alpha_3 \\ \lambda_1 & \lambda_2 & \lambda_3 \end{pmatrix}, \quad \beta_c = \begin{pmatrix} 0 & 0 & 0 \\ \mathbb{D}^t \beta_1 & \mathbb{D}^t \beta_2 & \mathbb{D}^t \beta_3 \end{pmatrix}.$$

Cela revient bien sûr à résoudre les trois systèmes séparément, mais la concaténation est utile dans la suite et simplifie l'écriture. En effet, une fois ce système résolu, on peut directement calculer la contribution de la cellule k à la matrice de rigidité globale

$$\mathbb{K}^k = \sum_{x_{T,k}} \frac{\omega_{x_{T,k}}}{\varepsilon} (\mathbb{J} \alpha_c)^t \mathbb{K}^{T,k} (\mathbb{J} \alpha_c) \quad \text{avec} \quad \mathbb{J} = \begin{pmatrix} I_{M_{mic}} & 0 \\ 0 & 0 \end{pmatrix}.$$

2.5.2 Matrice de contrainte

La matrice de contrainte est le point dur de la construction du système contraint. Elle contient les deux contraintes

1. $\int_{Y_\varepsilon(x_{T,k})} \tilde{w}_l - w_{l,lin} = 0$
2. $(\tilde{w}_l - w_{l,lin})(p) = (\tilde{w}_l - w_{l,lin})(p')$ pour tout couple (p, p') lié par les conditions périodiques **sans** redondance.

Cette matrice sera donc la forme

$$\mathbb{D} = \begin{pmatrix} b_1 & \\ \vdots & \tilde{\mathbb{D}} \\ b_{M_{mic}} & \end{pmatrix}$$

où $\tilde{\mathbb{D}} \in \mathcal{M}_{M_{mic}, C}(\mathbb{R})$ est la matrice qui force les conditions périodiques. On a

$$C = \underbrace{\text{Nbaretes}/2 + 1}_{\text{périodicité}} + \underbrace{1}_{f \dots = 0}$$

Elle est de la forme suivante pour notre numérotation des noeuds :

$$\tilde{\mathbb{D}}^t = \left(\begin{array}{c|c|c|c} \text{Liens entre angles} & \text{Liens horizontaux} & \text{Liens verticaux sur} & 0 \text{ pour les noeuds du} \\ (1) \leftrightarrow (2), (3) \leftrightarrow (4), & \text{sur les faces droite et} & \text{les faces haute et} & \text{centre} \\ (1) \leftrightarrow (4) & \text{gauche} & \text{basse} & \end{array} \right)$$

On peut montrer que pour la contrainte (1), on peut prendre

$$b_i = \int_{Y_\varepsilon} \psi_i.$$

3 Implémentation du code & résultats intermédiaires obtenus

Nous allons maintenant développer quelques éléments important à la compréhension du code. Celui-ci couvre la résolution de cas relativement étendus, A^ε pseudo-périodique avec des formes modulables, possibilité de faire des quadratures à un ou quatre points par cellule, parallélisation de la construction des matrices globales³. Cependant le code ne donne pas de bonne solution, nous allons donc détailler les quelques tests fait pour trouver le problème.

3.1 Présentation globale du code

Tout d’abord, pour une compréhension globale du code développé, on peut trouver en [Table 1](#) le détail de l’ensemble des fonctions utilisées et la structure du code simplifiée est présentée en [Figure 3](#).

Pour les micro-problèmes, on utilise le même maillage recentré au point de la quadrature, la matrice de contrainte n’a donc à être construite qu’une seule fois.

La boucle sur la construction est faite avec un `parfor` ce qui nous fait gagner en temps de calcul, cependant on ne peut pas directement répartir les contributions sur la matrice de rigidité car la parallélisation est rudimentaire sous MATLAB et ne peut pas gérer les problèmes de mutex complexes. Il faut donc sommer plus tard, et stocker en attendant.

Remarque 11. *Il faut enlever le `parfor` et mettre un `for` pour afficher / déboguer le code qui est dans la boucle.*

On prendra ε valant $H/10$, bien que la solution en soit indépendante (dans le code aussi semble-t-il à partir d’une certaine valeur).

3.2 Problèmes remarqués

On travaille dans un premier temps avec $A^\varepsilon(x, y) = I_2$, avec une quadrature à un seul point. Les projections de la macro base sur les micro-maillages semblent être trop basse comme on peut le voir en [Figure 1](#). On s’attendrait à des résultats dix fois plus grands me semble-t-il, mais je ne vois pas d’où le problème pourrait venir.

Les problèmes de cellules semblent être bien résolus dans le cas (i) *i.e.* $\tilde{w}_l = 0, \forall l \in [1, 3]$ pour tous les points de la quadrature, comme on peut le voir en [Figure 2](#), les résultats pourraient être bons pour les cas (iii) et (iv) puisqu’ils ressemblent qualitativement à ceux du TP3 (bien qu’il ne s’agisse pas exactement de la même quantité) ce qui est plus difficile à dire pour le cas (v).

Remarque 12. *Toutes ces fonctions de plot sont dans la fonction `calc_K_contrib`.*

Et la matrice qui en résulte semble respecter les conditions élémentaires

- Symétrique ;
- Somme des coefficients sur une ligne nulle.

Cependant, la matrice de rigidité qui en résulte a de très petits coefficients, et la solution globale est quasi constante à 10^{-4} , alors qu’on s’attendrait à retrouver la solution

$$u(x, y) = \sin(2\pi x) \sin(2\pi y).$$

On compare avec les coefficients obtenus dans le TP1, et on remarque que la matrice `Ke1` pour chaque matrice est cinq ordres de grandeurs trop petite, mais qu’il n’y a pas de coefficient constant liant le bon résultat et celui obtenu, de plus le signe trouvé pas toujours le bon.

³On pourrait penser qu’il s’agit d’un luxe, mais la parallélisation en MATLAB d’une boucle `for` est quasi immédiate et le temps gagné sur les tests est considérable.

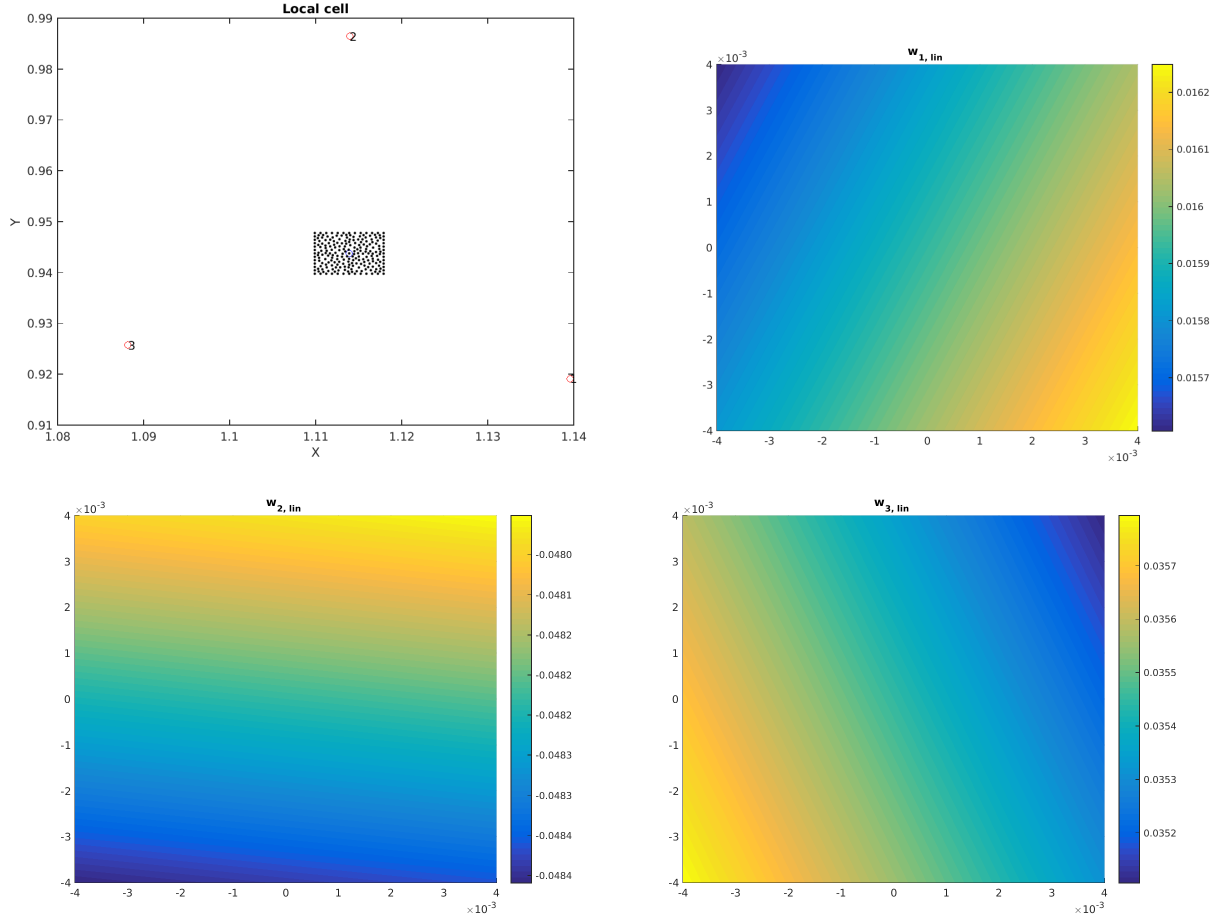


Figure 1: Projection des éléments de la macro-base sur les micro maillages. L'image de la cellule locale n'est pas très lisible, mais de gauche à droite on trouve les sommets 3, 2, 1 en rouge et le maillage est au centre. On voit en bleu le point de la quadrature.

Quand on prend $A^\varepsilon(x, y) = A^\varepsilon(y)$ comme dans les problèmes précédents, les solutions des problèmes de cellule

$$\tilde{w}_l - w_{l,lin}$$

ressemblent qualitativement à celles trouvées dans le TP3, mais sont quelques ordres de grandeurs en dessous... Cela pourrait venir de la contrainte sur les intégrales, la micro-matrice de rigidité semblant correspondre. En effet les coefficients de \mathbb{D} sont de l'ordre de l'unité, et les b_i sont de l'ordre de 10^{-7} .

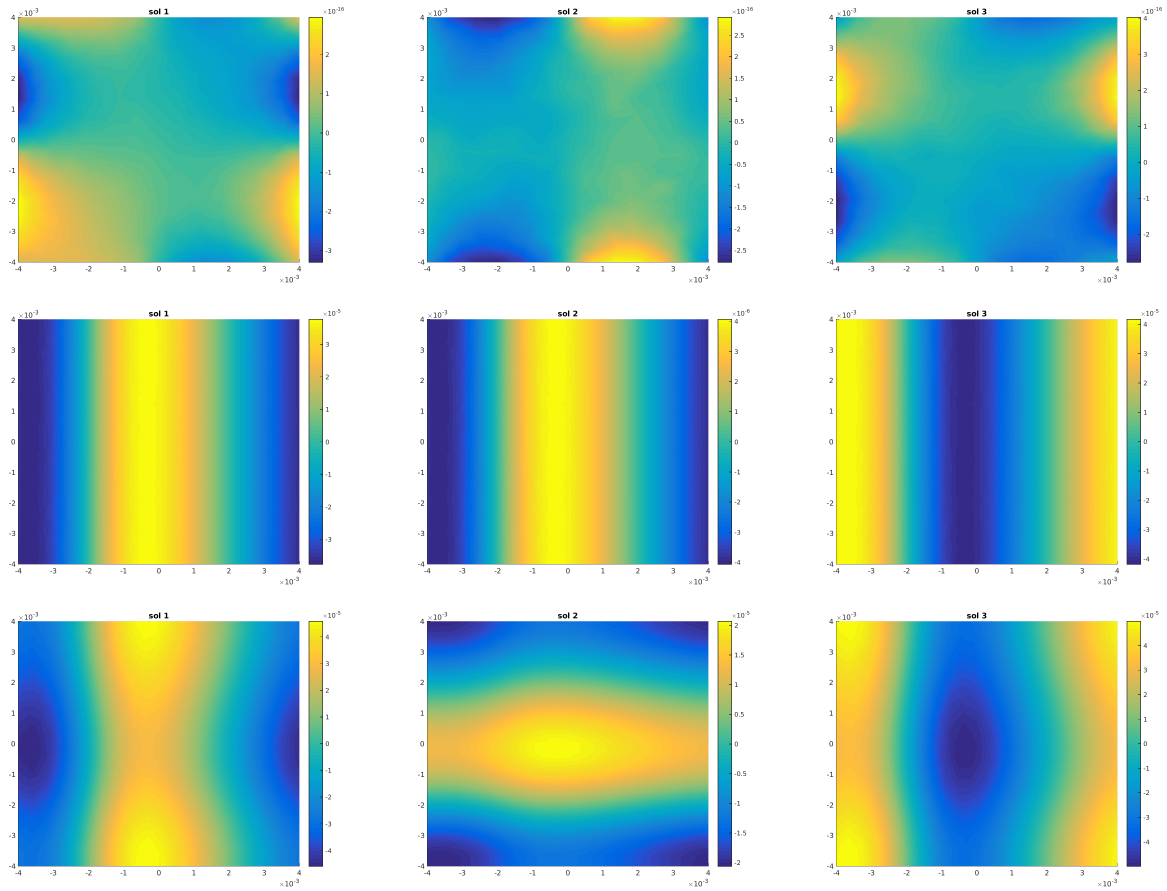


Figure 2: Solutions aux problèmes de cellule différentiel avec leur élément de la macro-base locale. Première ligne, cas (i) du TP3, seconde ligne cas (iii) et troisième ligne cas(v). Le cas (ii) donne les mêmes résultats que (i) et le cas (iv) donne le même résultat que (iii). Encore une fois la lisibilité n'est pas au rendez-vous, mais l'ordre de grandeur est de 10^{-6} pour le cas (i) et 10^{-5} pour les autres cas.

```

→ principal_dirichlet_aux
    • Lecture des maillages
→ calc_constr_mat
    • Constuction de la matrice de contraintes  $\mathbb{D}$ 
← calc_constr_mat
    • Pour macro-cellule dans le macro-maillage [PARALLELE]
        → calc_constr_mat
            • Calcul de la contribution de la cellule :
        → matK_elem
            • Pour point dans quadrature
                → calc_K_contrib
                    • Résoudre les trois problèmes modifiés
                ← calc_K_contrib
            • Fin Pour point dans quadrature
            • Sommer les valeurs de la quadrature
        ← matK_elem
            • Sauvegarder la contribution
            • ...
    • Fin Pour macro-cellule dans le macro-maillage
    • Pour macro-cellule dans le macro-maillage [SEQUENTIELLE]
        • Sommer les contributions des cellules pour construire les matrices de masse et rigidité
    • Fin Pour macro-cellule dans le macro-maillage
    • Résolution macro système
← principal_dirichlet_aux

```

Légende

- Opération classique
- Entrée dans une fonction
- ← Sortie d'une fonction
- Entrée dans une boucle
- [Fin ...] Sortie d'une boucle

Figure 3: Algorithme simplifié

Fonction	Usage
<code>affiche</code>	Affiche une fonction sur un maillage
<code>affichemaillage</code>	Affiche un maillage
<code>A</code>	Donne la valeur du tenseur A^ε en $x_{T,k}, x_{T,k}/\varepsilon$
<code>calc_constr_mat</code>	Calcule la matrice de contrainte \mathbb{D}
<code>calc_K_contrib</code>	Calcule la contribution d'un point de la quadrature
<code>f</code>	Retourne le membre de droite pour le problème macro-scopique
<code>lecture_msh</code>	Lecture d'un fichier <code>msh</code>
<code>matK_elem</code>	Calcule la contribution d'une cellule à la matrice de rigidité globale
<code>matK_elem_pbcell</code>	Calcule la contribution d'une cellule à la matrice de rigidité micro
<code>matM_elem</code>	Calcule la contribution d'une cellule à la matrice de masse globale
<code>parfor_progress</code>	Utilitaire pour afficher l'avancement (trouvé sur le net)
<code>principal_dirichlet_aux</code>	Fonction principale de la résolution du problème macro et constructions de toutes les matrices
<code>principal_dirichlet</code>	Script principal d'appel

Table 1: Récapitulatif des fonctions du code