

Rapport de stage 2A au Groupe Interdisciplinaire de  
Recherche en Éléments finis GIREF, Université Laval  
- Québec, CANADA

Grégoire MASSOT

15 Septembre 2015



**Dates du stage :** Du 22 Juin au 29 Août 2015 - 10 semaines  
**Tuteur de stage Université Laval :** André FORTIN  
**Tuteur École des Mines :** Julien BRUCHON

## Table des matières

<b>1</b>	<b>Résolution d'une EDP par éléments finis de type Nédélec</b>	<b>4</b>
1.1	Position du problème . . . . .	4
1.2	Formulation faible . . . . .	4
1.3	Maillage du domaine et approximation de Galerkin . . . . .	5
1.3.1	Espace d'approximation $V_h$ . . . . .	5
1.3.2	Base de l'espace $V_h$ . . . . .	5
1.3.3	Propriétés des fonctions de base . . . . .	5
1.4	Construction du système matriciel élémentaire . . . . .	5
1.4.1	Fonctions de base sur l'élément de référence . . . . .	6
1.4.2	Transformation géométrique . . . . .	6
1.4.3	Écriture des termes de formulation . . . . .	8
1.4.4	Intégration numérique . . . . .	9
1.5	Assemblage et résolution . . . . .	9
1.5.1	Assemblage . . . . .	9
1.5.2	Visualisation des résultats . . . . .	10
1.5.3	Étude de convergence $\mathcal{L}^2$ . . . . .	10
1.6	Remarques . . . . .	11
<b>2</b>	<b>Résolution d'une EDP par éléments finis de type Raviart-Thomas en 3D</b>	<b>12</b>
2.1	Position du problème . . . . .	12
2.2	Formulation faible . . . . .	12
2.3	Maillage du domaine et approximation de Galerkin . . . . .	12
2.4	Construction du système matriciel élémentaire . . . . .	13
2.4.1	Fonctions de base sur l'élément de référence . . . . .	13
2.4.2	Transformation géométrique . . . . .	14
2.4.3	Écriture des termes de formulation . . . . .	14
2.5	Assemblage et résolution . . . . .	15
2.5.1	Assemblage . . . . .	15
2.5.2	Visualisation des résultats . . . . .	15
2.5.3	Étude de convergence $\mathcal{L}^2$ . . . . .	16
<b>3</b>	<b>Aspects pratiques de l'utilisation de MEF++</b>	<b>17</b>
3.1	A propos de MEF++ . . . . .	17
3.2	Principes généraux de la résolution d'une EDP sous MEF++ . . . . .	17
3.3	Organisation du répertoire de travail . . . . .	17
3.3.1	Arborescence du répertoire de travail . . . . .	17
3.3.2	Dossier GIREF/ . . . . .	18
3.3.3	Dossier maillages/ . . . . .	19
3.3.4	Dossier Tests_prob***/ . . . . .	19
3.4	Logiciels auxiliaires utilisés . . . . .	20
3.4.1	Git . . . . .	20

3.4.2	Redmine . . . . .	20
3.4.3	Maple . . . . .	21
3.4.4	Outils de parallelisation . . . . .	21
3.4.5	Autres logiciels . . . . .	21
<b>4</b>	<b>Annexe : Démonstrations des transformations des fonctions de bases vers l'élément de référence</b>	<b>22</b>
4.1	Transformation des fonctions de bases de Nédélec . . . . .	22
4.2	Transformation du rotationnel des fonctions de bases de Nédélec . . . . .	22
4.2.1	En dimension 2 . . . . .	22
4.2.2	En dimension 3 . . . . .	23
4.3	Transformation des fonctions de bases de Raviart-Thomas . . . . .	23
4.4	Transformation de la divergence des fonctions de bases de Raviart-Thomas	24
<b>5</b>	<b>Références</b>	<b>25</b>

# 1 Résolution d'une EDP par éléments finis de type Nédélec

## 1.1 Position du problème

On travaille dans  $\mathbb{R}^d$  avec  $d = 2$  ou  $d = 3$ . On utilise le repère cartésien  $(\vec{e}_x, \vec{e}_y, \vec{e}_z)$  et les coordonnées cartésiennes usuelles  $(x, y, z)$ . Le domaine d'étude est noté  $\Omega \subset \mathbb{R}^3$ . Le problème que l'on cherche à résoudre sur  $\Omega$  est le suivant :

$$\vec{rot}(\vec{rot}(\vec{u}(x, y, z)) + \vec{u}(x, y, z) = \vec{f}(x, y, z)$$

où  $\vec{u}$  est le champ vectoriel que l'on cherche à déterminer et  $\vec{f}$  est un champ connu, appelé "source".

## 1.2 Formulation faible

On prend une fonction test  $\vec{v}(x, y, z)$  non identiquement nulle. On fait ensuite le produit scalaire de l'équation par  $\vec{v}(x, y, z)$  et on intègre sur  $\Omega$

$$\int_{\Omega} \vec{v} \cdot (\vec{rot}(\vec{rot}(\vec{u})) + \vec{u}) d\Omega = \int_{\Omega} \vec{v} \cdot \vec{f} d\Omega$$

On va supposer que la solution  $\vec{u}$  et la fonction test  $\vec{v}$  sont dans l'espace de Hilbert suivant :

$$V = H(rot, \Omega) = \{\vec{u} \in (\mathcal{L}^2(\Omega))^3 \text{ tel que } \vec{rot}(\vec{u}) \in (\mathcal{L}^2(\Omega))^3\}$$

On va simplifier le terme en  $\vec{rot}(\vec{rot}(\vec{u})) \cdot \vec{v}$ . Une formule d'analyse vectorielle nous donne

$$\vec{rot}(\vec{rot}(\vec{u})) \cdot \vec{v} = \text{div}(\vec{rot}(\vec{u}) \wedge \vec{v}) + \vec{rot}(\vec{u}) \cdot \vec{rot}(\vec{v})$$

En notant  $\wedge$  le produit vectoriel. Donc en intégrant sur le domaine  $\Omega$

$$\begin{aligned} \int_{\Omega} \vec{rot}(\vec{rot}(\vec{u})) \cdot \vec{v} d\Omega &= \int_{\Omega} \text{div}(\vec{rot}(\vec{u}) \wedge \vec{v}) + \vec{rot}(\vec{u}) \cdot \vec{rot}(\vec{v}) d\Omega \\ \Leftrightarrow \int_{\Omega} \vec{rot}(\vec{rot}(\vec{u})) \cdot \vec{v} d\Omega &= \int_{\partial\Omega} (\vec{rot}(\vec{u}) \wedge \vec{v}) \cdot \vec{n} d\Omega + \int_{\Omega} \vec{rot}(\vec{u}) \cdot \vec{rot}(\vec{v}) d\Omega \end{aligned}$$

Étant donné que  $(\vec{rot}(\vec{u}) \wedge \vec{v}) \cdot \vec{n} = (\vec{n} \wedge \vec{rot}(\vec{u})) \cdot \vec{v} = (\vec{v} \wedge \vec{n}) \cdot \vec{rot}(\vec{u})$ , le terme de bord va s'annuler si  $\vec{n} \wedge \vec{rot}(\vec{u}) = 0$  ou si  $\vec{v} \wedge \vec{n} = 0$ . La condition aux limites sera naturelle si  $\vec{n} \wedge \vec{rot}(\vec{u}) = 0$  sur le bord tandis qu'elle sera essentielle si  $\vec{v}$  est choisie de sorte que  $\vec{v} \wedge \vec{n} = 0$  sur le bord. Or  $\vec{rot}(\vec{u}) \wedge \vec{n} = 0$  et  $\vec{v} \cdot \vec{n} = 0$  sur le bord donc on a  $(\vec{rot}(\vec{u}) \wedge \vec{v}) \cdot \vec{n} = 0$  sur le bord  $\partial\Omega$ . On peut donc écrire la formulation variationnelle :

$$\int_{\Omega} \vec{rot}(\vec{u}) \cdot \vec{rot}(\vec{v}) + \vec{u} \cdot \vec{v} d\Omega = \int_{\Omega} \vec{v} \cdot \vec{f} d\Omega$$

La forme bilinéaire  $a(\vec{u}, \vec{v}) = \vec{rot}(\vec{u}) \cdot \vec{rot}(\vec{v}) + \vec{u} \cdot \vec{v}$  est coercive car elle correspond à la définition du produit scalaire dans  $H(rot, \Omega)$ . On peut donc appliquer le théorème de Lax-Milgram et avoir ainsi l'existence et l'unicité de la solution sur  $H(rot, \Omega)$ .

## 1.3 Maillage du domaine et approximation de Galerkin

### 1.3.1 Espace d'approximation $V_h$

On se place désormais dans le cas 2D. Soit un maillage à  $L$  triangles de notre domaine de travail  $\Omega$  avec des éléments triangles  $T^l$  numérotés par l'indice  $l \in \llbracket 1, L \rrbracket$  et de taille inférieure à  $h$ . On cherche la fonction  $\vec{u}$  dans l'espace d'approximation conforme  $V_h$  de  $H(\text{rot}, \Omega)$

$$V_h = \vec{v}_h \in H(\text{rot}, \Omega) \text{ tel que } \vec{v}_h|_{T^l} \in \mathcal{R}^1(T^l) \forall l \in \llbracket 1, L \rrbracket$$

avec  $\forall l \in \llbracket 1, L \rrbracket$

$$\mathcal{R}^1(T^l) = \{ \vec{v}_h(x, y) = \vec{a} + \gamma \begin{pmatrix} -y \\ x \end{pmatrix}, \gamma \in \mathbb{R}, \vec{a} \in \mathbb{R}^2 \}$$

### 1.3.2 Base de l'espace $V_h$

Pour chaque triangle  $l \in \llbracket 1, L \rrbracket$  de sommets  $S_1^l, S_2^l, S_3^l$ , on pose 3 fonctions [2]

$$\vec{\tau}_i^l(x, y) = \frac{1}{\det(S_{i+1}^l, \vec{S}_{i+2}^l, S_{i+1}^l, \vec{S}_i^l)} \begin{pmatrix} -y + y_{i+2}^l \\ x - x_{i+2}^l \end{pmatrix} \text{ sur } T^l, \vec{0} \text{ ailleurs}$$

Les  $\vec{\tau}_i^l$  forment une base de l'espace d'approximation conforme  $V_h$ . On se concentre désormais sur un seul élément triangle d'indice  $l = e$  noté  $K_e$ . Ce triangle a ses arêtes orientées.

On peut alors écrire le champ inconnu  $\vec{u}$  à l'intérieur de l'élément sous la forme de la combinaison linéaire des 3 champs vectoriels  $\vec{\tau}_1(x, y) = \vec{\tau}_1^e(x, y)$ ,  $\vec{\tau}_2(x, y) = \vec{\tau}_2^e(x, y)$  et  $\vec{\tau}_3(x, y) = \vec{\tau}_3^e(x, y)$ .

$$\vec{u}(x, y) = u_1 \vec{\tau}_1(x, y) + u_2 \vec{\tau}_2(x, y) + u_3 \vec{\tau}_3(x, y)$$

### 1.3.3 Propriétés des fonctions de base

Ces fonctions ont la propriété suivante :

$$\int_{A_i} \vec{\tau}_j(x, y) \cdot \vec{t}_i ds = \begin{cases} 1 & \text{si } i=j \\ 0 & \text{sinon} \end{cases}$$

avec  $ds$  l'élément de longueur et  $\vec{t}_i$  le vecteur unitaire tangent à l'arête  $A_i$ . Les fonctions de bases sont donc associées à une arête particulière. Sur les 2 autres arêtes du triangle le champ est normal à l'arête. On peut donc écrire que pour  $i \in \llbracket 1, 3 \rrbracket$  :

$$u_i = \int_{A_i} \vec{u}(x, y) \cdot \vec{t}_i ds$$

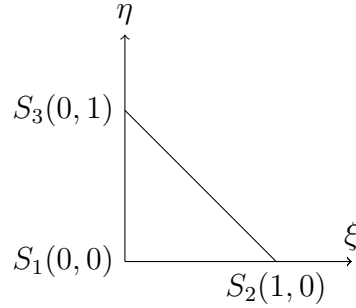
Le coefficient  $u_i$  est égal à l'intégrale le long de l'arête  $A_i$  du produit scalaire  $\vec{u}(x, y) \cdot \vec{t}_i$

## 1.4 Construction du système matriciel élémentaire

On va maintenant exprimer nos fonctions de base sur l'élément courant à partir des fonctions de base sur l'élément de référence afin de pouvoir ramener toutes les intégrales qui constituent le système matriciel élémentaire à des intégrales sur l'élément de référence.

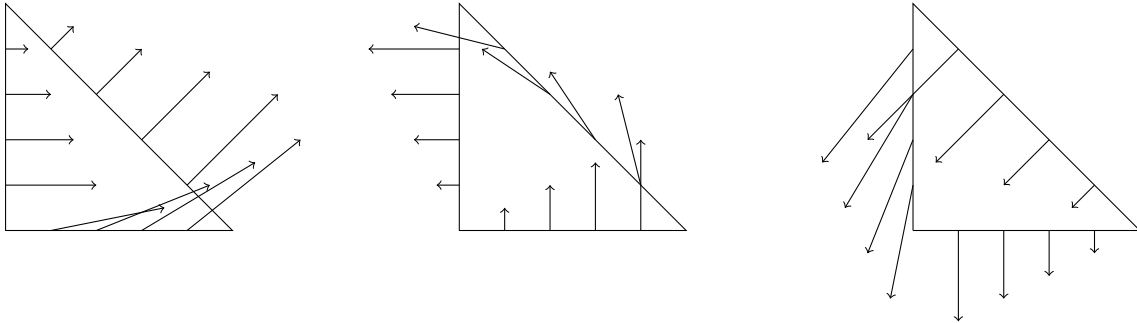
### 1.4.1 Fonctions de base sur l'élément de référence

On se place désormais sur l'élément de référence triangle  $\hat{\Omega}$  avec le système de coordonnées  $(\xi, \eta)$ . Les sommets du triangle ont pour coordonnées  $S_1(\xi = 0, \eta = 0)$ ,  $S_2(\xi = 1, \eta = 0)$ ,  $S_3(\xi = 0, \eta = 1)$



Sur l'élément de référence, les fonctions de base s'écrivent de la façon suivante :

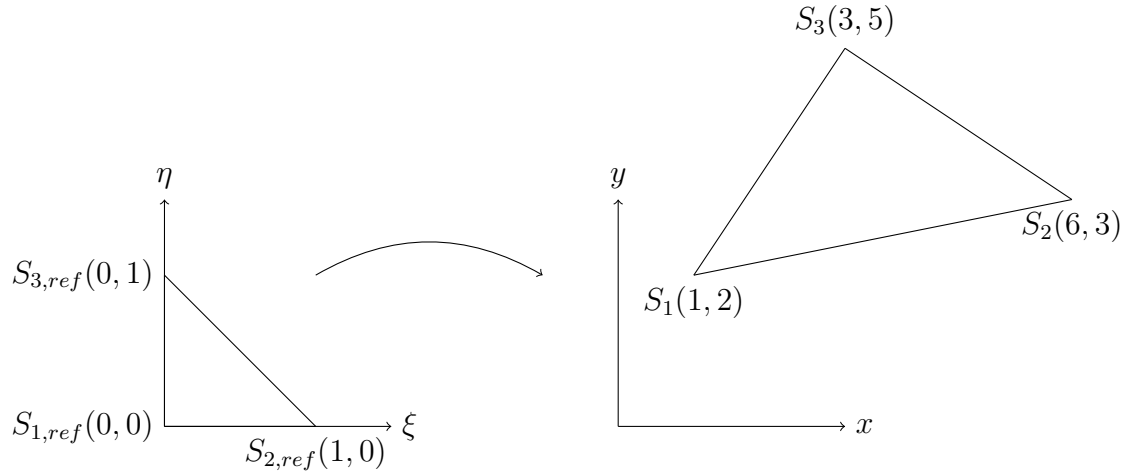
$$\begin{aligned}\vec{\tau}_1^{ref}(x, y) &= (1 - \eta, \xi) \\ \vec{\tau}_2^{ref}(x, y) &= (-\eta, \xi) \\ \vec{\tau}_3^{ref}(x, y) &= (-\eta, \xi - 1)\end{aligned}$$



Les fonctions  $\vec{\tau}_1(x, y)$ ,  $\vec{\tau}_2(x, y)$  et  $\vec{\tau}_3(x, y)$  sur l'élément de référence

### 1.4.2 Transformation géométrique

On a explicité les fonctions de bases sur un élément de référence. Il s'agit maintenant de savoir comment on passe de cet élément de référence à un élément ayant des sommets de coordonnées quelconques et des arêtes d'orientations quelconques.



On définit une fonction

$$\begin{aligned}\Psi : \quad \mathbb{R}^2 &\rightarrow \mathbb{R}^2 \\ (\xi, \eta) &\rightarrow (x, y)\end{aligned}$$

qui à chaque sommet  $i$  de l'élément de référence associe le sommet  $i$  de l'élément courant. On peut aussi voir cette fonction comme un champ vectoriel sur l'ensemble du maillage. Il est appelé dans MEF++ le "Champ de transformation géométrique". On va chercher ici une transformation linéaire, c'est à dire que l'on écrit :

$$\begin{aligned}x &= a_1\xi + b_1\eta + c_1 \\ y &= a_2\xi + b_2\eta + c_2\end{aligned}$$

On résout ce système de 6 équations à 6 inconnues :

$$\begin{cases} f(S_{1,ref}) = S_1 \\ f(S_{2,ref}) = S_2 \\ f(S_{3,ref}) = S_3 \end{cases}$$

On obtient alors :

$$\begin{cases} a_1 = x_2 - x_1 \\ b_1 = x_3 - x_1 \\ c_1 = x_1 \\ a_2 = y_2 - y_1 \\ b_2 = y_3 - y_1 \\ c_2 = y_1 \end{cases}$$

avec

$$2\alpha = (x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1)$$

On en déduit la matrice jacobienne de  $\Psi$

$$J = \begin{pmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{pmatrix} = \begin{pmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \end{pmatrix}$$

On prend alors les formules suivantes pour la transformation des fonctions de base. Pour  $i \in \llbracket 1, 3 \rrbracket$

$$\begin{aligned}\vec{\tau}_i(x(\xi, \eta), y(\xi, \eta)) &= (J^{-1})^t \vec{\tau}_i^{ref}(\xi, \eta) \\ \vec{rot}(\vec{\tau}_i)(x(\xi, \eta), y(\xi, \eta)) &= \frac{1}{\det(J)} \vec{rot}(\vec{\tau}_i^{ref})(\xi, \eta)\end{aligned}$$

Celles ci sont démontrées dans l'annexe.

### 1.4.3 Écriture des termes de formulation

La fonction inconnue  $\vec{u}$  et la fonction test  $\vec{v}$  s'écrivent dans notre espace d'approximation interne  $V_h$

$$\begin{aligned}\vec{u}(x, y) &= u_1 \vec{\tau}_1(x, y) + u_2 \vec{\tau}_2(x, y) + u_3 \vec{\tau}_3(x, y) \\ \vec{v}(x, y) &= v_1 \vec{\tau}_1(x, y) + v_2 \vec{\tau}_2(x, y) + v_3 \vec{\tau}_3(x, y)\end{aligned}$$

avec

$$\begin{aligned}\vec{\tau}_1(x, y) &= \tau_{1,x}(x, y) \vec{e}_x + \tau_{1,y}(x, y) \vec{e}_y \\ \vec{\tau}_2(x, y) &= \tau_{2,x}(x, y) \vec{e}_x + \tau_{2,y}(x, y) \vec{e}_y \\ \vec{\tau}_3(x, y) &= \tau_{3,x}(x, y) \vec{e}_x + \tau_{3,y}(x, y) \vec{e}_y\end{aligned}$$

On écrit ces fonctions sous forme matricielle

$$\begin{aligned}\vec{u}(x, y) &= \begin{pmatrix} \tau_{1,x} & \tau_{2,x} & \tau_{3,x} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} \vec{e}_x + \begin{pmatrix} \tau_{1,y} & \tau_{2,y} & \tau_{3,y} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} \vec{e}_y \\ \vec{v}(x, y) &= \begin{pmatrix} \tau_{1,x} & \tau_{2,x} & \tau_{3,x} \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} \vec{e}_x + \begin{pmatrix} \tau_{1,y} & \tau_{2,y} & \tau_{3,y} \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} \vec{e}_y \\ &= \begin{pmatrix} v_1 & v_2 & v_3 \end{pmatrix} \begin{pmatrix} \tau_{1,x} \\ \tau_{2,x} \\ \tau_{3,x} \end{pmatrix} \vec{e}_x + \begin{pmatrix} v_1 & v_2 & v_3 \end{pmatrix} \begin{pmatrix} \tau_{1,y} \\ \tau_{2,y} \\ \tau_{3,y} \end{pmatrix} \vec{e}_y\end{aligned}$$

On écrit de même les rotationnels sous forme matricielle

$$\begin{aligned}\vec{rot}(\vec{u}(x, y)) &= u_1 \vec{rot}(\vec{\tau}_1(x, y)) + u_2 \vec{rot}(\vec{\tau}_2(x, y)) + u_3 \vec{rot}(\vec{\tau}_3(x, y)) \\ &= \begin{pmatrix} \left(\frac{\partial \tau_{1,y}}{\partial x} - \frac{\partial \tau_{1,x}}{\partial y}\right) & \left(\frac{\partial \tau_{2,y}}{\partial x} - \frac{\partial \tau_{2,x}}{\partial y}\right) & \left(\frac{\partial \tau_{3,y}}{\partial x} - \frac{\partial \tau_{3,x}}{\partial y}\right) \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} \vec{e}_z \\ \vec{rot}(\vec{v}(x, y)) &= \begin{pmatrix} \left(\frac{\partial \tau_{1,y}}{\partial x} - \frac{\partial \tau_{1,x}}{\partial y}\right) & \left(\frac{\partial \tau_{2,y}}{\partial x} - \frac{\partial \tau_{2,x}}{\partial y}\right) & \left(\frac{\partial \tau_{3,y}}{\partial x} - \frac{\partial \tau_{3,x}}{\partial y}\right) \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} \vec{e}_z \\ &= \begin{pmatrix} v_1 & v_2 & v_3 \end{pmatrix} \begin{pmatrix} \left(\frac{\partial \tau_{1,y}}{\partial x} - \frac{\partial \tau_{1,x}}{\partial y}\right) \\ \left(\frac{\partial \tau_{2,y}}{\partial x} - \frac{\partial \tau_{2,x}}{\partial y}\right) \\ \left(\frac{\partial \tau_{3,y}}{\partial x} - \frac{\partial \tau_{3,x}}{\partial y}\right) \end{pmatrix} \vec{e}_z\end{aligned}$$



La source  $\vec{f}$  s'écrit :

$$\vec{f}(x, y) = f_x(x, y)\vec{e}_x + f_y(x, y)\vec{e}_y$$

On revient à la formulation faible et on divise par  $(v_1 \ v_2 \ v_3)$ . On obtient le système linéaire à 3 inconnues :

$$M^e U^e = F^e$$

avec

- $M^e$  la matrice masse  $3 \times 3$
- $F^e$  le vecteur résidu
- $U^e$  le vecteur que l'on cherche

$$\begin{aligned} M_{i,j}^e &= \int_{K_e} \vec{\tau}_i \cdot \vec{\tau}_j + r \vec{ot}(\vec{\tau}_i) \cdot r \vec{ot}(\vec{\tau}_j) dK_e \\ &= \int_{\hat{\Omega}} ((J^{-1})^t \tau_i^{ref} \cdot (J^{-1})^t \tau_j^{ref} + \frac{1}{\det(J)} r \vec{ot}(\tau_i^{ref}) \cdot \frac{1}{\det(J)} r \vec{ot}(\tau_j^{ref})) \det(J) d\hat{\Omega} \\ &= \int_{\hat{\Omega}} \tilde{M}_{i,j} d\hat{\Omega} \\ F_i^e &= \int_{K_e} \vec{f}_i \cdot \vec{\tau}_i dK_e \\ &= \int_{\hat{\Omega}} (\vec{f}_i \cdot (J^{-1})^t \tau_i^{ref}) \det(J) d\hat{\Omega} \\ &= \int_{\hat{\Omega}} \tilde{F}_i d\hat{\Omega} \end{aligned}$$

#### 1.4.4 Intégration numérique

Enfin, on calcule les intégrales en faisant une quadrature de Hammer. On intègre des polynômes d'ordre maximum 2. On choisit donc un nombre  $p$  de points d'intégrations supérieur à 3 :

$$\begin{aligned} M_{i,j} &= \int_{\hat{\Omega}} \tilde{M}_{i,j} d\hat{\Omega} = \sum_{k=1}^p W_k \tilde{M}_{i,j}(\xi_k, \eta_k) \\ F_i &= \int_{\hat{\Omega}} \tilde{F}_i d\hat{\Omega} = \sum_{k=1}^p W_k \tilde{F}_i(\xi_k, \eta_k) \end{aligned}$$

### 1.5 Assemblage et résolution

#### 1.5.1 Assemblage

On va ensuite créer ce système élémentaire pour chaque élément (ici des triangles) du maillage. Il s'agit ensuite de superposer les systèmes linéaires élémentaires pour obtenir notre système linéaire global.

On commence par numéroter les arêtes dans le maillage global  $k = 1..n$  et les orienter afin de les identifier. Le triangle d'indice  $l$  a les arêtes d'indices  $k_{l1}, k_{l2}, k_{l3}$ . Le terme de la matrice de masse élémentaire  $M_{i,j}^e$  va être ajouté dans la case  $M_{k_{li}, k_{lj}}$  et celui du vecteur résidu élémentaire  $F_i^e$  va être ajouté à la case  $F_{k_{li}}$

### 1.5.2 Visualisation des résultats

On fait le test sur un carré avec les sommets de coordonnées  $(0, 0)$ ,  $(0, 1)$ ,  $(1, 1)$ ,  $(1, 0)$ . On utilise la solution analytique suivante

$$\vec{u}(x, y) = \begin{pmatrix} 2\sin(\pi x)\cos(\pi y) \\ \cos(\pi x)\sin(\pi y) \end{pmatrix}$$

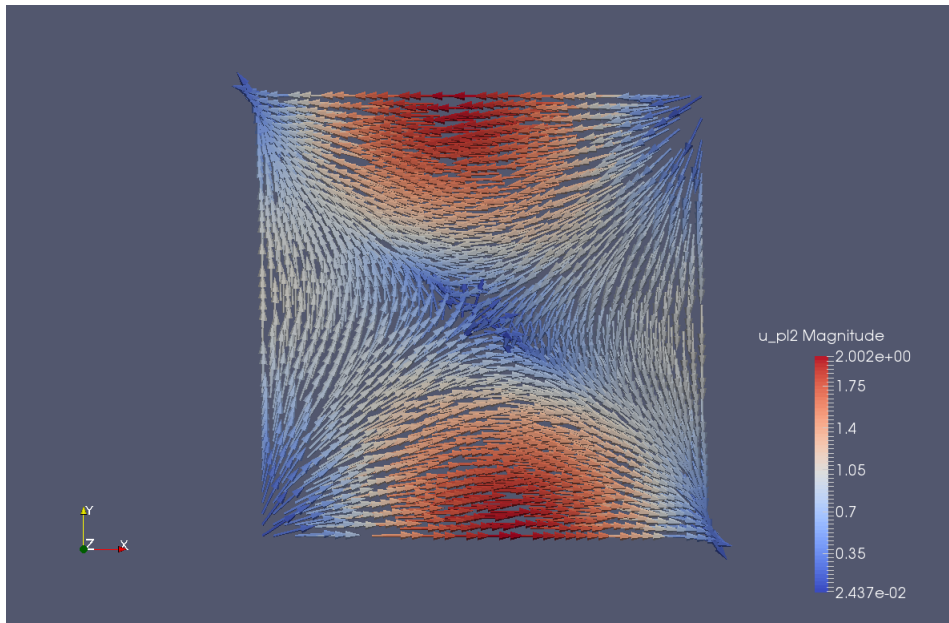
Ce qui conduit à la source suivant :

$$\vec{f}(x, y) = \begin{pmatrix} (\pi^2 + 2)\sin(\pi x)\cos(\pi y) \\ (1 - \pi^2)\cos(\pi x)\sin(\pi y) \end{pmatrix}$$

Après résolution, on obtient un vecteur solution de taille le nombre  $n$  d'arêtes. Celui ci n'est pas exploitable en lui-même pour visualiser la solution. Il faut, pour chaque élément, calculer le champ sur quelques points de l'élément en reprenant la formule :

$$\vec{u}(x, y) = u_{k_{l1}}\vec{\tau}_{k_{l1}}(x, y) + u_{k_{l2}}\vec{\tau}_{k_{l2}}(x, y) + u_{k_{l3}}\vec{\tau}_{k_{l3}}(x, y) \text{ sur le triangle } l$$

On fait donc une réinterpolation d'un champ `v3dnelelec` vers un champ `Vectoriel3D` dans `MEF++`.



Solution obtenue par éléments finis

### 1.5.3 Étude de convergence $\mathcal{L}^2$

On veut calculer la norme  $\mathcal{L}^2$  de la différence entre le vecteur solution éléments finis et le vecteur solution analytique.

$$e_{\mathcal{L}^2} = \sqrt{\int_{\Omega} (u - u_{sol})^2 d\Omega}$$

Il faut pour cela que la solution analytique donnée plus haut soit comparable à la solution éléments finis. Il faut donc faire une réinterpolation du champ `Vectoriel3D` vers `v3dnedelec` dans `MEF++` avec la formule :

$$u_i^{sol} = \int_{A_i} \vec{u}(x, y) \cdot \vec{t}_i ds$$

On obtient alors le tableau de convergence suivant.

$h$	$e_{\mathcal{L}^2}(h)$
$5 \cdot 10^{-2}$	0.64
$2.5 \cdot 10^{-2}$	0.32
$1.25 \cdot 10^{-2}$	0.16

On a donc une convergence à l'ordre 1 de  $\vec{u}$

## 1.6 Remarques

- Le nombre de degrés de libertés est égal au nombre d'arêtes dans le maillage. C'est pour cela que l'on appelle les éléments de Nédélec des éléments d'arêtes.
- On peut généraliser au cas 3D avec des éléments tétraèdres. On a alors 6 arêtes par éléments. On a toujours une convergence de  $\vec{u}$  à l'ordre 1
- L'intérêt des éléments de Nédélec est de pouvoir imposer facilement une condition au bord  $\vec{u} \wedge \vec{n} = 0$ , c'est à dire imposer un champ normal au bord. Il suffit d'imposer que les coefficients associés aux arêtes au bord soient nuls. C'est l'équivalent d'une condition de Dirichlet homogène pour les éléments de Lagrange.

## 2 Résolution d'une EDP par éléments finis de type Raviart-Thomas en 3D

Les éléments de Raviart-Thomas sont aux faces ce que les éléments de Nédélec sont aux arêtes.

### 2.1 Position du problème

On travaille dans  $\mathbb{R}^3$ , on utilise le repère cartésien  $(\vec{e}_x, \vec{e}_y, \vec{e}_z)$  et les coordonnées cartésiennes usuelles  $(x, y, z)$ . Le domaine d'étude est noté  $\Omega \subset \mathbb{R}^3$ . Le problème que l'on cherche à résoudre sur  $\Omega$  est le suivant :

$$\begin{cases} \vec{p} &= -\vec{grad}(u) \\ \text{div}(\vec{p}) &= f \end{cases}$$

$\vec{p}$  et  $u$  sont nos champs inconnus. On a un *problème mixte*.

### 2.2 Formulation faible

On prend les fonctions test  $\vec{q}(x, y, z)$  et  $v(x, y, z)$  non identiquement nulles. On fait ensuite le produit scalaire de la première équation par  $\vec{q}$ , on multiplie la seconde par  $v$  et on intègre sur  $\Omega$

$$\begin{cases} \int_{\Omega} \vec{p} \cdot \vec{q} + \vec{grad}(u) \cdot \vec{q} d\Omega = 0 \\ \int_{\Omega} v \text{div}(\vec{p}) d\Omega = \int_{\Omega} v f d\Omega \end{cases}$$

On va supposer que la solution  $u$  et sa fonction test  $v$  sont dans l'espace de Hilbert  $V = \mathcal{L}^2(\Omega)$  et que  $\vec{p}$  et  $\vec{q}$  sont dans l'espace de Hilbert :

$$Q = H(\text{div}, \Omega) = \{\vec{p} \in (\mathcal{L}^2(\Omega))^3 \text{ tel que } \text{div}(\vec{p}) \in \mathcal{L}^2(\Omega)\}$$

On écrit la formulation variationnelle :

$$\begin{cases} \int_{\Omega} \vec{p} \cdot \vec{q} - u \text{div}(\vec{q}) d\Omega = 0 \forall \vec{q} \in Q \\ \int_{\Omega} v \text{div}(\vec{p}) d\Omega = \int_{\Omega} v f d\Omega \forall v \in V \end{cases}$$

Cette formulation variationnelle implique que  $u = 0$  sur  $\partial\Omega$ . En effet le théorème de la divergence nous donne

$$\int_{\Omega} \vec{grad}(u) \cdot \vec{q} d\Omega = \int_{\partial\Omega} u \vec{q} \cdot \vec{n} d\partial\Omega - \int_{\Omega} u \text{div}(\vec{q}) d\Omega$$

Ceci montre que la condition de Dirichlet  $u = 0$  devient une condition naturelle pour la formulation variationnelle donnée.

### 2.3 Maillage du domaine et approximation de Galerkin

Soit un maillage à  $L$  tétraèdres de notre domaine de travail  $\Omega$  avec des éléments tétraèdres numérotés par l'indice  $l \in \llbracket 1, L \rrbracket$  et de taille inférieure à  $h$ . On se concentre désormais sur un seul élément tétraèdre d'indice  $l = e$  noté  $K_e$ . Ce tétraèdre a ses faces orientées.

On peut alors écrire le champ inconnu  $\vec{u}$  à l'intérieur de l'élément sous la forme de la combinaison linéaire des 4 champs vectoriels  $\vec{\tau}_1(x, y, z) = \vec{\tau}_1^e(x, y, z)$ ,  $\vec{\tau}_2(x, y, z) = \vec{\tau}_2^e(x, y, z)$ ,  $\vec{\tau}_3(x, y, z) = \vec{\tau}_3^e(x, y, z)$  et  $\vec{\tau}_4(x, y, z) = \vec{\tau}_4^e(x, y, z)$

$$\vec{p}(x, y, z) = p_1 \vec{\tau}_1(x, y, z) + p_2 \vec{\tau}_2(x, y, z) + p_3 \vec{\tau}_3(x, y, z) + p_4 \vec{\tau}_4(x, y, z)$$

Ces fonctions ont la propriété suivante :

$$\int_{F_i} \vec{\tau}_j(x, y, z) \cdot \vec{n}_i dS = \begin{cases} 1 & \text{si } i=j \\ 0 & \text{sinon} \end{cases}$$

avec  $\vec{n}_i$  le vecteur unitaire normal à la face  $F_i$  et  $dS$  l'élément d'aire de la face. Les fonctions de bases sont donc associées à une face particulière. Sur les 3 autres faces du triangle le champ est tangent à la face. On peut donc écrire que pour  $i \in \llbracket 1, 4 \rrbracket$  :

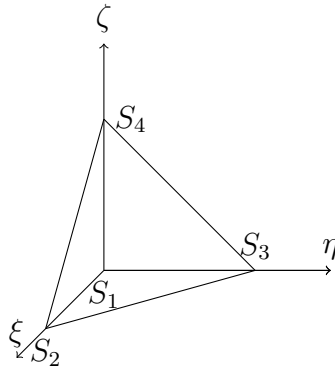
$$p_i = \int_{F_i} \vec{p}(x, y, z) \cdot \vec{n}_i(x, y, z) dS$$

Le coefficient  $p_i$  est égal au flux sur la face  $F_i$  de  $\vec{p}(x, y, z)$

## 2.4 Construction du système matriciel élémentaire

### 2.4.1 Fonctions de base sur l'élément de référence

On se place désormais sur l'élément de référence tétraèdre  $\hat{\Omega}$  avec le système de coordonnées  $(\xi, \eta, \zeta)$ . Les sommets du tétraèdre ont pour coordonnées  $S_1(\xi = 0, \eta = 0, \zeta = 0)$ ,  $S_2(\xi = 1, \eta = 0, \zeta = 0)$ ,  $S_3(\xi = 0, \eta = 1, \zeta = 0)$ ,  $S_4(\xi = 0, \eta = 0, \zeta = 1)$ .



Sur l'élément de référence, les fonctions de base s'écrivent de la façon suivante :

$$\begin{aligned} \vec{\tau}_1^{ref}(x, y, z) &= (\xi, \eta, \zeta - 1) \\ \vec{\tau}_2^{ref}(x, y, z) &= (\xi, \eta - 1, \zeta) \\ \vec{\tau}_3^{ref}(x, y, z) &= (\xi, \eta, \zeta) \\ \vec{\tau}_4^{ref}(x, y, z) &= (\xi - 1, \eta, \zeta) \end{aligned}$$

### 2.4.2 Transformation géométrique

On généralise la transformation  $\Psi$  et sa matrice jacobienne  $J$  de la partie 1 à  $\mathbb{R}^3$ . On a alors les formules suivantes pour les transformations des fonctions de base du tétraèdre courant vers le tétraèdre de référence :

$$\begin{aligned}\vec{\tau}_i(x(\xi, \eta, \zeta), y(\xi, \eta, \zeta), z(\xi, \eta, \zeta)) &= \frac{J}{\det(J)} \vec{\tau}_i^{ref}(\xi, \eta, \zeta) \\ \text{div}(\vec{\tau}_i)(x(\xi, \eta, \zeta), y(\xi, \eta, \zeta), z(\xi, \eta, \zeta)) &= \frac{1}{\det(J)} \text{div}(\vec{\tau}_i^{ref})(\xi, \eta, \zeta)\end{aligned}$$

Ces formules sont démontrées dans l'annexe.

### 2.4.3 Écriture des termes de formulation

La fonction inconnue  $\vec{p}$  et la fonction test  $\vec{q}$  s'écrivent dans notre espace d'approximation interne  $V_h$

$$\begin{aligned}\vec{p}(x, y) &= p_1 \vec{\tau}_1(x, y, z) + p_2 \vec{\tau}_2(x, y, z) + p_3 \vec{\tau}_3(x, y, z) + p_4 \vec{\tau}_4(x, y, z) \\ \vec{q}(x, y) &= q_1 \vec{\tau}_1(x, y, z) + q_2 \vec{\tau}_2(x, y, z) + q_3 \vec{\tau}_3(x, y, z) + q_4 \vec{\tau}_4(x, y, z)\end{aligned}$$

La fonction inconnue  $u$  et sa fonction test  $v$  sont constants sur l'élément. Après simplification de la première équation par  $(q_1 \ q_2 \ q_3)$ , et de la seconde équation par  $v$ , on obtient le système linéaire à 5 inconnues suivant :

$$\begin{pmatrix} A^e & -(B^e)^t \\ B^e & 0 \end{pmatrix} \begin{pmatrix} p \\ u \end{pmatrix} = \begin{pmatrix} 0 \\ f \end{pmatrix}$$

avec

- $A^e$  une matrice carrée  $4 \times 4$
- $B^e$  un vecteur ligne  $1 \times 4$
- $f^e$  un scalaire  $1 \times 1$

$$\begin{aligned}A_{i,j}^e &= \int_{K_e} \vec{\tau}_i \cdot \vec{\tau}_j dK_e \\ &= \int_{\hat{\Omega}} \frac{J}{\det(J)} \tau_{i,ref} \cdot \frac{J}{\det(J)} \tau_{j,ref} \det(J) d\hat{\Omega} \\ &= \int_{\hat{\Omega}} \tilde{A}_{i,j} d\hat{\Omega} \\ B_{1,i}^e &= \int_{K_e} \text{div}(\vec{\tau}_i) dK_e \\ &= \int_{\hat{\Omega}} \frac{1}{\det J} \text{div}(\tau_i^{ref}) \det(J) d\hat{\Omega} \\ &= \int_{\hat{\Omega}} \tilde{B}_i d\hat{\Omega}\end{aligned}$$

Comme dans la partie 1, on calcule ces intégrales avec une quadrature de Hammer. On intègre des polynômes d'ordre maximum 2. On prend donc au minimum  $p = 4$  points d'intégration.

## 2.5 Assemblage et résolution

### 2.5.1 Assemblage

On assemble la matrice  $A$  comme on assemblerait la matrice  $M$  dans la partie 1 par analogie entre les arêtes et les faces. Pour chaque tétraèdre  $i \in \llbracket 1, n \rrbracket$  et ses 4 faces  $F_{i_j}$  d'indice global  $i_j, j \in \llbracket 1, 4 \rrbracket$  on pose :

$$\begin{cases} B_{i,i_j} = B_{1,j}^e \\ \text{puis 0 pour le reste de la ligne } i \end{cases}$$

### 2.5.2 Visualisation des résultats

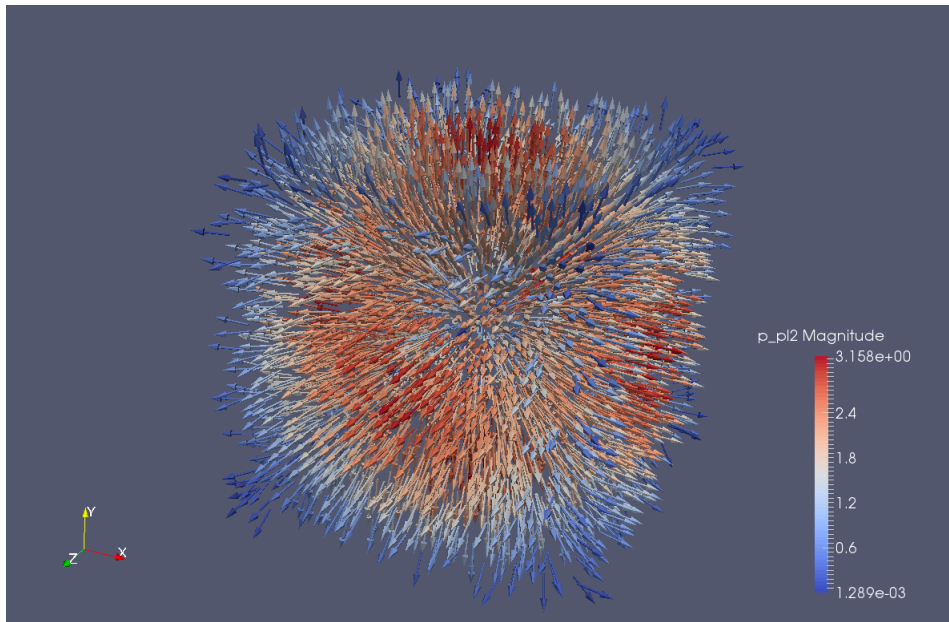
On fait le test sur un cube de sommets  $(0, 0, 0)$ ,  $(0, 1, 0)$ ,  $(1, 1, 0)$ ,  $(1, 0, 0)$ ,  $(0, 0, 1)$ ,  $(0, 1, 1)$ ,  $(1, 1, 1)$ ,  $(1, 0, 1)$ . On utilise la solution analytique suivante :

$$\begin{aligned} u(x, y, z) &= \sin(\pi x) \sin(\pi y) \sin(\pi z) \\ \vec{p}(x, y, z) &= \begin{pmatrix} -\pi \cos(\pi x) \sin(\pi y) \sin(\pi z) \\ -\pi \sin(\pi x) \cos(\pi y) \sin(\pi z) \\ -\pi \sin(\pi x) \sin(\pi y) \cos(\pi z) \end{pmatrix} \end{aligned}$$

Ce qui conduit au terme source suivant :

$$f(x, y, z) = 3\pi^2 \sin(\pi x) \sin(\pi y) \sin(\pi z)$$

Pour pouvoir visualiser le champ  $\vec{p}$  obtenu par éléments finis, on fait une réinterpolation d'un champ `v3draviart` vers un champ `Vectoriel3D`



Solution  $\vec{p}$  obtenue par éléments finis

### 2.5.3 Étude de convergence $\mathcal{L}^2$

Pour comparer la solution analytique  $p_{sol}^{\vec{}}$  à la solution obtenue par éléments finis, on fait un réinterpolation de  $p_{sol}^{\vec{}}$  d'un champ `Vectoriel3D` vers un champ `v3draviart`.

$h$	$e_{\vec{p}, \mathcal{L}^2}(h)$	$e_{u, \mathcal{L}^2}(h)$
$25 \cdot 10^{-2}$	0.57	0.10
$12.5 \cdot 10^{-2}$	0.31	0.053
$6.125 \cdot 10^{-2}$	0.17	0.027

On a donc une convergence à l'ordre 1 en  $\vec{p}$  et  $u$



## 3 Aspects pratiques de l'utilisation de MEF++

### 3.1 A propos de MEF++

Le logiciel MEF++ est développé depuis 1996 par le GIREF, une unité du département de mathématiques de l'Université Laval. Il est développé principalement par des ingénieurs et des professeurs d'université. Il est aussi développé par des doctorants et des stagiaires. MEF++ est développé à la fois pour la recherche publique et la recherche privée. Le laboratoire est donc financé à la fois par le CRSNG (un organisme public canadien) et par des entreprises.

MEF++ est principalement développé en C++. Il compte à ce jour 1.7 millions de lignes de code

### 3.2 Principes généraux de la résolution d'une EDP sous MEF++

Le choix de l'EDP à résoudre est fait à l'intérieur du code MEF++. Si on veut changer de problème, on change d'exécutable. En revanche le choix de  $\Omega$ , des conditions aux limites et des valeurs des champs connus sont lus par MEF++ depuis des fichiers externes.

On cherche à résoudre les EDP sur des objets quelconques. Par exemple un cube, une poutre, un tore. On commence par représenter cet objet avec une *géométrie*. Ce sont les points, les courbes, les surfaces et les volumes qui permettent de se représenter l'objet dans l'espace. Les points, les courbes, les surfaces et les volumes sont regroupés sous la notion de *lieux géométriques*. La *géométrie* est donc l'ensemble des *lieux géométriques* de notre objet. La géométrie est enregistrée dans un fichier `.geo`

On définit le *maillage* à partir de la *géométrie*. Il peut y avoir plusieurs *maillages* par *géométrie* mais un *maillage* ne correspond qu'à une seule *géométrie*. Les *maillages* sont enregistrés au format `.mail`.

On définit ensuite les *entités géométriques*. Ce sont des regroupements de *lieux géométriques*. On définit souvent plusieurs *entités géométriques* par *géométrie*. Elles servent principalement à préciser sur quels bords on impose les conditions aux limites ou à définir les propriétés physiques des milieux. Ils sont enregistrés sous le format `.enti`.

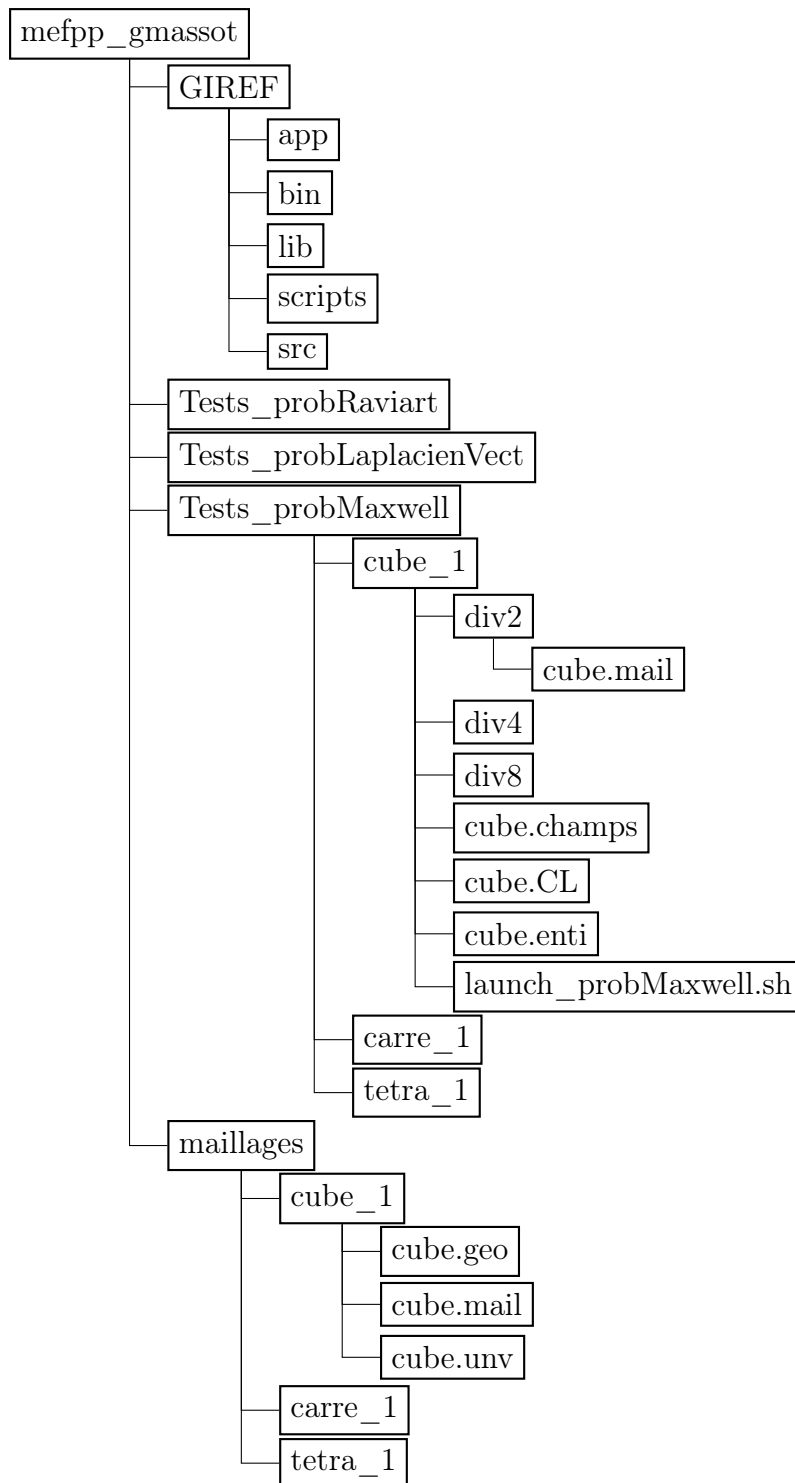
Les informations sur les champs qui sont en jeu dans l'EDP et les propriétés physiques des milieux sont enregistrées dans un fichier `.champ`

Les conditions aux limites sont définies dans le fichier `.CL`. Elles font références aux `.enti` et aux `.champs`.<sup>3</sup>

On crée le `.geo` à la main ou avec GMSH. On le passe dans GMSH pour obtenir un maillage au format `.unv`. On passe le `.unv` dans iMEF++ pour créer les entités géométriques et obtenir un `.mail` et un `.enti` On code à la main le `.champs` et le `.CL` ou on les récupère à partir de templates.

### 3.3 Organisation du répertoire de travail

#### 3.3.1 Arborescence du répertoire de travail



Arborescence du répertoire de travail personnel `mefpp_gmassot`

### 3.3.2 Dossier GIREF/

Le dossier `GIREF/` contient le programme MEF++. C'est la copie du *dépôt* Git. Il est composé de plusieurs dossiers :

- `app/` : contient des fichiers sources `.cc` ne contenant que des fonctions `main`. Une fonction `main` correspond à un exécutable et à une EDP particulière à résoudre.
- `bin/` : contient les exécutables issus de ces fonctions `main`

- **lib/** : contient les bibliothèques dynamiques **.so**. Ce sont ces outils qui sont appelés dans les fonctions main du dossier **app/**
- **src/** : contient les sources dont sont issus les **.so**
- **scripts/** : contient les scripts **.py** et **.sh** de routine (mise en forme des résultats exportés par exemple)

### 3.3.3 Dossier maillages/

Le dossier **maillages/** contient les différents dossiers de maillages. Ce sont les dossiers de travail pour créer un **.unv** à partir d'un **.geo** avec GMSH. On fait en sorte d'avoir des maillages grossiers en sortie de GMSH pour pouvoir les raffiner automatiquement avec l'outil **GIREF/bin/diviseAreteEn2** et ainsi faire des études de convergence.

### 3.3.4 Dossier Tests\_prob\*\*\*/

On crée pour chaque problème un nouveau fichier main dans **GIREF/app/** et on crée un nouveau dossier générique dans la racine **mefpp\_gmassot** nommé **Tests\_prob\*\*\*/**. Par exemple **Tests\_probMaxwell/** correspond au premier problème étudié en partie 1, **Tests\_probRaviart/** correspond au second problème étudié en partie 2.

Dans chacun de ces dossiers, on copie les maillages du dossier **maillages/** qui nous intéressent, on crée des fichiers **.champs** et **.CL** et on lance ensuite des scripts shell pour avoir une série de maillages de plus en plus fins et effectuer de manière automatique des tests de convergence sur une EDP.

On commence par créer des sous dossiers **div2/**, **div4/**, **div8/...** Pour chaque sous-dossier **div\$i/**, on lance **GIREF/bin/diviseAreteEn2** et on met le résultat dans **../div\$i+1/**. On crée ensuite, dans chacun des sous-dossiers, un lien symbolique vers les fichiers **.enti**, **.CL**, **.champs**.

```
#!/bin/bash
mkdir div2
/home/mefpp_gmassot/GIREF/bin/diviseAreteEn2.opt cube div2/cube
for i in 2 4 8 16 32
do
mkdir -p div$i
cd div$i
ln -sf ../cube.{enti,CL,champs} .
cd ..
done

for i in 2 4 8 16
do
let "j = 2*i"
cd div$i
/home/mefpp_gmassot/GIREF/bin/diviseAreteEn2.opt cube ../div$j/cube
cd ..
done
```

On peut ainsi lancer le programme à tester dans chaque dossier (par exemple **probMaxwell.opt**)

```

for i in 2 4 8 16
do
cd div$i
/home/mefpp_gmassot/GIREF/bin/probMaxwell.opt cube
cd ..
done

```

On affiche dans chaque dossier les résultats importants qui ont été exportés par MEF++. Par exemple l'erreur en norme  $\mathcal{L}^2$  entre la solution analytique et la solution éléments finis.

```

for i in 2 4 8 16
do
cd div$i
cat NormeResiduL2.valeurs
cd ..
done

```

## 3.4 Logiciels auxiliaires utilisés

### 3.4.1 Git

Git est l'outil de gestion de version utilisé au GIREF. L'ensemble des fichiers sources, *le dépôt*, est stocké sur un serveur. On peut faire une copie du *dépôt*, le modifier, puis faire un *commit*, c'est à dire renvoyer les fichiers modifiés sur le serveur. Voici les principales commandes utilisées :

Commande Git	Fonction
<code>git status</code>	Permet de voir l'ensemble des fichiers modifiés sur la copie du <i>dépôt</i> mais non commités. En vert les fichiers qui sont sur la liste du prochain <i>commit</i> . En rouge les fichiers qui n'y sont pas
<code>git diff fichier</code>	Permet de voir les modifications faites sur <i>fichier</i> depuis le dernier <i>commit</i>
<code>git add fichier</code>	Ajoute <i>fichier</i> sur la liste du prochain <i>commit</i>
<code>git checkout (-p) fichier</code>	Enlève les modifications faites en local sur <i>fichier</i>

On utilise aussi qGit qui est une interface graphique pour GIT. Elle permet de naviguer facilement entre les différents commits.

### 3.4.2 Redmine

Redmine est le gestionnaire de projet de MEF++. C'est un site web en interne qui contient, entre autres :

- Une documentation : Elle est générée automatiquement à partir des commentaires des fichiers sources
- Un wiki
- Un fil d'actualité
- Un forum

### 3.4.3 Maple

On utilise la méthode des solutions manufacturières pour tester les programmes. On a une solution connue de l'EDP et on en déduit le terme source à partir de cette EDP. On utilise alors Maple afin de faire le calcul formel et rentrer l'expression du terme source dans le fichier `.champs` Par exemple :

```
with(VectorCalculus)
Vx := 3*cos(Pi*x)*sin(Pi*y)*sin(Pi*z)
Vy := 2*sin(Pi*x)*cos(Pi*y)*sin(Pi*z)
Vz := sin(Pi*x)*sin(Pi*y)*cos(Pi*z)
v := VectorField('<,>'(Vx, Vy, Vz), 'cartesian'[x, y, z])
champs := simplify(-Laplacian(v)+v)
```

### 3.4.4 Outils de parallélisation

Le code MEF++ est parallélisé. On peut accélérer l'exécution des programmes en les lançant avec `mpirun`. On peut aussi les faire exécuter sur le centre de calcul de l'université *Colosse*.

On peut aussi paralléliser la compilation des bibliothèques sur l'ensemble des machines du laboratoire avec l'outil `icecream`. C'est utile quand on a tous les `.so` à recompiler.

### 3.4.5 Autres logiciels

Parmi les autres logiciels utilisés, on a :

- GMSH : c'est un mailleur libre et gratuit : [www.geuz.org/gmsh/](http://www.geuz.org/gmsh/)
- Paraview : Il permet de visualiser les résultats obtenus sur la géométrie en 2D et en 3D
- Vim : L'éditeur de texte pour modifier rapidement les fichiers `.champs`, les `.CL` et les scripts
- QtCreator : un IDE complet et gratuit pour coder en C++
- iMEF++ : un logiciel annexe à MEF++. Il sert principalement à créer des fichiers `.mail` et `.enti` à partir du fichier `.unv` issu de GMSH

## 4 Annexe : Démonstrations des transformations des fonctions de bases vers l'élément de référence

On reprend la transformation  $\Psi$  et sa matrice jacobienne  $J$ . On écrit la formule de Nanson [1] qui nous servira par la suite. Elle donne la transformation d'un élément d'aire orienté  $\vec{N}dA$  de  $\hat{\Omega}$  en  $\vec{n}da$  de  $\Omega$  par  $\Psi$  :

$$\vec{n}da = \det(J)(J^{-1})^t \vec{N}dA$$

On travaille en dimension  $n = 2$  ou  $n = 3$ . On note  $(x_1, \dots, x_n)$  les coordonnées de l'espace courant et  $(\hat{x}_1, \dots, \hat{x}_n)$  les coordonnées de l'espace de référence.

### 4.1 Transformation des fonctions de bases de Nédelec

Soit  $\mathcal{C}$  un contour quelconque et  $\hat{\mathcal{C}}$  sa transformation dans l'espace  $\hat{\Omega}$ . On cherche un champ  $u^{\vec{ref}}$  tel que :

$$\int_{\mathcal{C}} \vec{u} \cdot d\vec{r} = \int_{\hat{\mathcal{C}}} u^{\vec{ref}} \cdot d\vec{R}$$

On explicite le champ  $\vec{u} \cdot d\vec{r}$  :

$$\begin{aligned} \vec{u} \cdot d\vec{r} &= \sum_{j=1}^n u_j dx_j \\ &= \sum_{j=1}^n \sum_{i=1}^n u_j \frac{\partial x_j}{\partial \hat{x}_i} d\hat{x}_i \\ &= \sum_{i=1}^n \left( \sum_{j=1}^n u_j \frac{\partial x_j}{\partial \hat{x}_i} \right) d\hat{x}_i \\ &= \sum_{i=1}^n u_i^{\vec{ref}} d\hat{x}_i \end{aligned}$$

On en déduit que  $u_i^{\vec{ref}} = \sum_{j=1}^n u_j \frac{\partial x_j}{\partial \hat{x}_i}$  puis que  $u^{\vec{ref}} = J^t \vec{u}$ . On en déduit ainsi que :

$$\boxed{\vec{u}(x_1, \dots, x_n) = (J^{-1})^t u^{\vec{ref}}(\hat{x}_1, \dots, \hat{x}_n)}$$

### 4.2 Transformation du rotationnel des fonctions de bases de Nédelec

#### 4.2.1 En dimension 2

On écrit le flux de  $\vec{rot}(\vec{u})$  à travers  $\Omega$ . On utilise la formule de Green pour se ramener à une circulation sur le bord  $\partial\Omega$  :

$$\iint_{\Omega} \vec{rot}(\vec{u}) \cdot \vec{e}_z d\Omega = \int_{\partial\Omega} \vec{u} \cdot d\vec{r}$$

On fait de même avec la fonction  $u^{\vec{ref}}$

$$\iint_{\hat{\Omega}} \vec{rot}(u^{\vec{ref}}) \cdot \vec{e}_z d\hat{\Omega} = \int_{\partial\hat{\Omega}} u^{\vec{ref}} \cdot d\vec{R}$$

On utilise l'égalité des circulations de  $\vec{u}$  sur  $\mathcal{C} = \partial\Omega$  et de  $u^{\vec{ref}}$  sur  $\hat{\mathcal{C}} = \partial\hat{\Omega}$  du 4.1 pour écrire :

$$\iint_{\Omega} r\vec{ot}(\vec{u}) \cdot \vec{e}_z d\Omega = \iint_{\hat{\Omega}} r\vec{ot}(u^{\vec{ref}}) \cdot \vec{e}_z d\hat{\Omega}$$

On prend le membre de gauche et on se ramène à une intégrale sur  $\hat{\Omega}$  avec la formule  $d\Omega = \det(J)d\hat{\Omega}$  :

$$\iint_{\hat{\Omega}} \det(J) r\vec{ot}(\vec{u}) \cdot \vec{e}_z d\hat{\Omega} = \iint_{\hat{\Omega}} r\vec{ot}(u^{\vec{ref}}) \cdot \vec{e}_z d\hat{\Omega}$$

Ceci est vrai pour tout  $\hat{\Omega}$ . On en déduit la formule de transformation entre  $r\vec{ot}(\vec{u})$  et  $r\vec{ot}(u^{\vec{ref}})$

$$\boxed{r\vec{ot}(\vec{u})(x_1, x_2) = \frac{1}{\det(J)} r\vec{ot}(u^{\vec{ref}})(\hat{x}_1, \hat{x}_2)}$$

### 4.2.2 En dimension 3

On écrit le flux de  $r\vec{ot}(\vec{u})$  à travers  $\mathcal{S} \in \partial\Omega$ . On utilise la formule de Stokes pour se ramener à une circulation sur un contour  $\mathcal{C}$  sur lequel s'appuie  $\mathcal{S}$  :

$$\iint_{\mathcal{S}} r\vec{ot}(\vec{u}) \cdot \vec{n} da = \int_{\mathcal{C}} \vec{u} \cdot d\vec{r}$$

On fait de même avec la fonction  $u^{\vec{ref}}$

$$\iint_{\hat{\mathcal{S}}} r\vec{ot}(u^{\vec{ref}}) \cdot \vec{N} dA = \int_{\hat{\mathcal{C}}} u^{\vec{ref}} \cdot d\vec{R}$$

On utilise l'égalité des circulations de  $\vec{u}$  sur  $\mathcal{C}$  et de  $u^{\vec{ref}}$  sur  $\hat{\mathcal{C}}$  du 4.1 pour écrire :

$$\iint_{\mathcal{S}} r\vec{ot}(\vec{u}) \cdot \vec{n} da = \iint_{\hat{\mathcal{S}}} r\vec{ot}(u^{\vec{ref}}) \cdot \vec{N} dA$$

On prend le membre de gauche et on se ramène à une intégrale sur  $\hat{\mathcal{S}}$  avec la formule de Nanson : en remarquant que  $\vec{u} \cdot ((J^{-1})^t \vec{N}) = (J^{-1} \vec{u}) \cdot \vec{N}$  :

$$\iint_{\hat{\mathcal{S}}} J^{-1} \det(J) r\vec{ot}(\vec{u}) \cdot dA \vec{N} = \iint_{\hat{\mathcal{S}}} r\vec{ot}(u^{\vec{ref}}) \cdot dA \vec{N}$$

Ceci est vrai pour tout  $\hat{\mathcal{S}}$ . On en déduit la formule de transformation :

$$\boxed{r\vec{ot}(\vec{u})(x_1, x_2, x_3) = \frac{J}{\det(J)} r\vec{ot}(u^{\vec{ref}})(\hat{x}_1, \hat{x}_2, \hat{x}_3)}$$

## 4.3 Transformation des fonctions de bases de Raviart-Thomas

On se place désormais seulement dans le cas 3D. Soit  $\mathcal{S}$  une surface quelconque et  $\hat{\mathcal{S}}$  sa transformation dans l'espace  $\hat{\Omega}$ . On cherche un champ  $u^{\vec{ref}}$  tel que :

$$\iint_{\mathcal{S}} \vec{u} \cdot \vec{n} da = \iint_{\hat{\mathcal{S}}} u^{\vec{ref}} \cdot \vec{N} dA$$

On développe le terme de gauche avec la formule de Nanson en remarquant que  $\vec{u} \cdot ((J^{-1})^t \vec{N}) = (J^{-1} \vec{u}) \cdot \vec{N}$  :

$$\iint_{\hat{\mathcal{S}}} \vec{u} \cdot \det(J) (J^{-1})^t \vec{N} dA = \iint_{\hat{\mathcal{S}}} u^{\vec{ref}} \cdot \vec{N} dA$$

. On en déduit que :

$$\iint_{\hat{\mathcal{S}}} \left( \det(J) J^{-1} \vec{u} - u^{\vec{ref}} \right) \cdot \vec{N} dA = 0$$

Ceci est vrai pour toute surface  $\hat{\mathcal{S}}$ , donc

$$\boxed{\vec{u}(x_1, x_2, x_3) = \frac{J}{\det(J)} u^{\vec{ref}}(\hat{x}_1, \hat{x}_2, \hat{x}_3)}$$

#### 4.4 Transformation de la divergence des fonctions de bases de Raviart-Thomas

On écrit l'intégrale sur le volume  $\Omega$  de  $\text{div}(\vec{u})$ . On utilise le théorème de la divergence pour se ramener à un flux sur le bord  $\partial\Omega$  :

$$\iiint_{\Omega} \text{div}(\vec{u}) d\Omega = \iint_{\partial\Omega} \vec{u} \cdot \vec{n} da$$

On fait de même avec la fonction  $u^{\vec{ref}}$

$$\iiint_{\hat{\Omega}} \text{div}(u^{\vec{ref}}) d\hat{\Omega} = \iint_{\partial\hat{\Omega}} u^{\vec{ref}} \cdot \vec{N} dA$$

On a alors, d'après l'égalité de départ de 4.3 :

$$\iiint_{\Omega} \text{div}(\vec{u}) d\Omega = \iiint_{\hat{\Omega}} \text{div}(u^{\vec{ref}}) d\hat{\Omega}$$

On transforme le membre de gauche en utilisant la formule  $d\Omega = \det(J) d\hat{\Omega}$  :

$$\iiint_{\hat{\Omega}} \det(J) \text{div}(\vec{u}) d\hat{\Omega} = \iiint_{\hat{\Omega}} \text{div}(u^{\vec{ref}}) d\hat{\Omega}$$

Ceci est vrai pour tout volume  $\hat{\Omega}$ . On en déduit la formule de transformation de la divergence :

$$\boxed{\text{div}(\vec{u})(x_1, x_2, x_3) = \frac{1}{\det(J)} \text{div}(u^{\vec{ref}})(\hat{x}_1, \hat{x}_2, \hat{x}_3)}$$



## 5 Références

- [1] André GARON André FORTIN. *Les éléments finis : de la théorie à la pratique*, pages 293–294. 2011.
- [2] Éric LUNÉVILLE Patrick CIARLET. *La méthode des éléments finis de la théorie à la pratique II. Compléments*, pages 127–130. Les Presses de L’ENSTA, 2010.