Méthode des éléments finis

Ibrahim ALAME

ESTP

19/10/2022

Éléments finis d'Hermite

φ_i des éléments finis de Lagrange est construite pour être continue d'un élément à l'autre, mais pas sa dérivée...

Un élément fini d'Hermite est un triplet (K, Σ, P) tel que :

- K est un élément géométrique de $\mathbb{R}^n$, compact, connexe, et d'intérieur non vide ;
- $\Sigma = \{\sigma_1, \cdots, \sigma_N\}$ un ensemble de N formes linéaires sur l'espace des fonctions définies sur K, ou sur un sous-espace plus régulier contenant P ;
- P est un espace vectoriel de dimension finie de fonctions réelles définies sur K, et tel que Σ soit $P-$unisolvant.

Opérateur de $P-$interpolation

- Un opérateur de $P-$interpolation sur Σ est un opérateur Π qui à toute fonction v définie sur K associe la fonction Πv de P définie par :

$$\Pi v = \sum_{i=1}^{N} \sigma_i(v)\varphi_i$$

- Πv est l'unique élément de P qui prend les mêmes valeurs que v sur les points de Σ.
-
$$\sigma_j(\varphi_i) = \delta_{ij}, \quad 1 \leq i,j \leq N$$

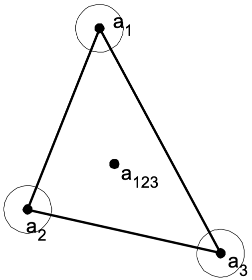
- Suivant les éléments utilisés, ces fonctions de base pourront être de classe C^1 ou même plus, et il en sera donc de même pour la solution approchée u_h.

Éléments unidimensionnels

Élément	cubique	quintique
K	segment $[a\,;b]$	segment $[a\,;b]$
Σ	$\{p(a), p'(a), p(b), p'(b)\}$	$\{p(a), p'(a), p''(a), p(b), p'(b), p''(b)\}$
P	P_3	P_3
Régularité	C^1 et H^2	C^2 et H^3

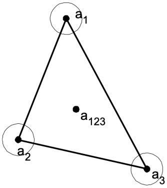
Élément	
K	triangle de sommets $\{a_1, a_2, a_3\}$
Σ	$\left\{p(a_i), \frac{\partial p}{\partial x}(a_i), i = 1, 2, 3\right\} \cup \{p(a_0)\}$
P	P_3
Régularité	C^0, mais pas C^1

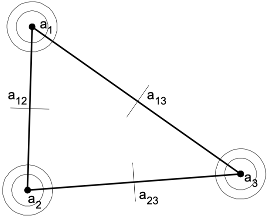
Élément triangle

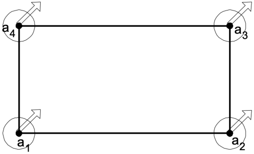


Élément	
K	triangle de sommets $\{a_1, a_2, a_3\}$
Σ	$\left\{p(a_i), \frac{\partial p}{\partial x}(a_i), i = 1,2,3\right\} \cup \{p(a_0)\}$
P	P_3
Régularité	C^0, mais pas C^1

Éléments bidimensionnels







Élément	
K	triangle de sommets $\{a_1, a_2, a_3\}$
Σ	$\left\{p(a_i), \frac{\partial p}{\partial x}(a_i), \frac{\partial p}{\partial y}(a_i), \frac{\partial^2 p}{\partial x^2}(a_i), \frac{\partial^2 p}{\partial y^2}(a_i), \frac{\partial^2 p}{\partial x \partial y}(a_i), i = 1, 2, 3\right\} \cup \left\{\frac{\partial p}{\partial n}(a_{ij}), 1 \leqslant i < j \leqslant 3\right\}$
P	P_5
Régularité	C^1

Élément	Q_3
K	rectangle de sommets $\{a_1, a_2, a_3, a_4\}$ de côtés parallèles aux axes
Σ	$\left\{p(a_i), \frac{\partial p}{\partial x}(a_i), \frac{\partial p}{\partial y}(a_i), \frac{\partial^2 p}{\partial x \partial y}(a_i), i = 1, \ldots, 4\right\}$
P	P_3
Régularité	C^1

Les fonctions de base

- $\Phi_i(x)$ les fonctions de base associées aux valeurs nodales de la fonction v_i.
- $\Psi_i(x)$ les fonctions de base associées aux valeurs nodales de la dérivée $(\frac{dv}{dx})_i$,

$$\Pi v(x) = \sum_{i=1}^{n} v_i \Phi_i(x) + \sum_{i=1}^{n} (\frac{dv}{dx})_i \Psi_i(x)$$

Sur un élément $[x_1, x_2]$, cette approximation s'écrit :

$$v^h(x) = v_1 \Phi_1(x) + (\frac{dv}{dx})_1 \Psi_1(x) + v_2 \Phi_2(x) + (\frac{dv}{dx})_2 \Psi_2(x)$$

Φ_i et Ψ_i vérifient les conditions :

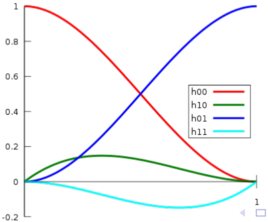
$$\left\{ \begin{array}{ll} \Phi_i(x_j) = \delta_{ij} & \Phi_i'(x_j) = 0 \\ \Psi_i(x_j) = 0 & \Psi_i'(x_j) = \delta_{ij} \end{array} \right.$$

Les fonctions de base

$$\begin{cases} \Phi_i(x) = [1 - 2(x - x_i)\lambda_i'(x_i)]\,\lambda_i^2(x) \\ \Psi_i(x) = (x - x_i)\lambda_i^2(x) \end{cases}$$

- élément de référence de longueur h : $x_1 = 0$ et $x_2 = h$,
- $\lambda_1(x) = 1 - \frac{x}{h}$ et $\lambda_2(x) = \frac{x}{h}$. On pose $\xi = \frac{x}{h}$:

$$\begin{cases} \Phi_1(x) = (1 + 2\xi)(1 - \xi)^2 \\ \Phi_2(x) = (3 - 2\xi)\xi^2 \\ \Psi_1(x) = h\xi(1 - \xi)^2 \\ \Psi_2(x) = h(\xi - 1)\xi^2 \end{cases}$$



Liste des sommets

- N_s nombre des sommets,
- Pour chaque sommet $i = 1, \cdots, N_s$,
  - les coordonnées du sommet i.
- Exemples en dimension 1 : $a = x_0 < x_1 < x_2 < \cdots < x_n = b$
  - $N_s = n + 1$
  - Tableau à une entrée de longueur N_s

i	1	2	...	N_s
x	x_0	x_1	...	x_n

Il ne faut pas toujours stocker cette structure de données. Si la grille est uniforme, elle est donnée de façon implicite par

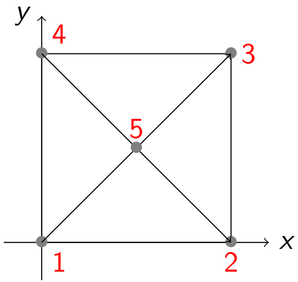
$$N_s = n + 1; \; h = \frac{1}{n} \quad x_0 = a; \; x_i = a + x_0 + ih, \; i = 1, \cdots, (N_s - 1).$$

Liste des sommets en dimension 2

Tableau des coordonnées $\{(x_{1i}, x_{2i})\}_{i=1}^{N_s}$.

- Nombre de sommets : N_s
- Tableau à deux lignes et N_s colonnes

i	1	2	...	N_s
$\mathbf{x}_{1i}$	x_{11}	x_{12}	...	x_{1n}
$\mathbf{x}_{2i}$	x_{21}	x_{22}	...	x_{2n}



i	1	2	3	4	5
$\mathbf{x}_{1i}$	0	L	L	0	$L/2$
$\mathbf{x}_{2i}$	0	0	H	H	$H/2$
Γ	1	1	1	1	0

Table de connectivité

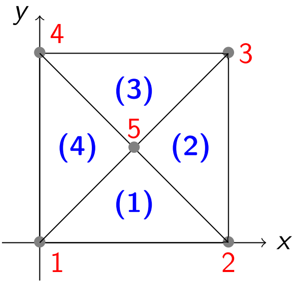
- N_e nombre d'éléments,
- Pour chaque élément $e = 1, \cdots, N_e$,
  - $n_{e,j}$: numéro du sommet j de l'élément e.
- Exemple en dimension 1 :
  - $N_e = N_s - 1$
  - Pour chaque élément $e = 1, \cdots, N_e$,

$$n_{e,1} = e, \quad n_{e,2} = e + 1$$

Pour le cas mono-dimensionnel, la table de connectivité est implicite à partir de la liste des sommets. Il est inutile par conséquent de la stocker.

Table de connectivité en dimension 2

- N_e nombre de triangles,
- $n_{e,j}\,:\, e = 1,\cdots,N_e,\; j = 1,2,3$



e	1	2	3	4
1	5	5	3	1
2	1	2	4	5
3	2	3	5	4

Table de connectivité en dimension 2

L'ordre dans lequel sont donnés les numéros de sommet n'est pas important. Si on peut, il faut respecter un sens comme ici le sens trigonométrique. Cela peut faciliter pour certains problèmes quelques points de programmation.
Souvent on ajoute un numéro de référence pour pouvoir introduire une caractérisation des équations au niveau de chaque élément.

e	1	2	3	4
1	5	5	3	1
2	1	2	4	5
3	2	3	5	4
4	1	2	1	2

les éléments 1 et 3 ont les mêmes caractéristiques, de même que les éléments 2 et 4 .

L'assemblage

Soit $K^{[e]}$ la matrice de rigidité de l'élément e. La matrice globale du système $K^{\#}$ est donnée par

$$\left[K^{\#}\right]_{ij} = \sum_{K^{[e]} \in \mathscr{T}^h, n_{[e]i_e}=i, n_{[e]j_e}=j} \left[K^{[e]}\right]_{i_e j_e}$$

Pour former $K^{\#}$, il suffit de parcourir les éléments en ajoutant successivement la contribution de chaque de chaque matrice élémentaire $\left[K^{[e]}\right]_{i_e j_e}$ à la matrice totale $\left[K^{\#}\right]_{ij}$ avec $i = n_{[e]i_e}$, $j = n_{[e]j_e}$.

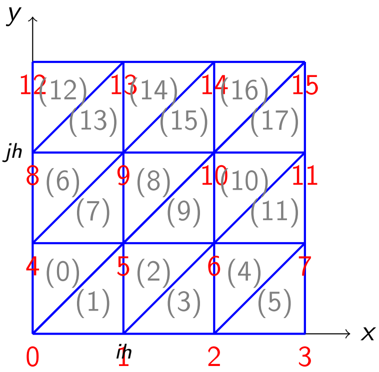
Algorithm 1 Algorithme général schématisant un assemblage

$K^{\#} := 0$; {Initialisation des éléments de $K^{\#}$ stockés en mémoire}
do $e = 1, N_E$ {Boucle sur les éléments}
  $K_{\text{elem}} := \text{form}(e)$; {Formation de la matrice élémentaire}
  do $i_e = 1, \ell$ {Boucle sur les lignes de la matrice élémentaire}
    $i := n(e, i_e)$; {Indice de ligne du coefficient à incrémenter}
    do $j_e = 1, \ell$ {Boucle sur les colonnes de la matrice élémentaire}
      $j := n(e, j_e)$; {Indice de colonne du coefficient à incrémenter}
      $K^{\#}(i, j) := K^{\#}(i, j) + K_{\text{elem}}(i_e, j_e)$; {Assemblage}
    enddo
  enddo
enddo

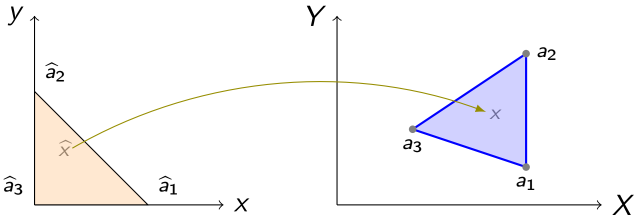
$$(\mathscr{P}_v) \begin{cases} \text{Trouver } u \in V \text{ vérifiant} \\ a(u,v) = \ell(v) \quad \forall v \in V \end{cases}$$

où

$$a(u,v) = \int_\Omega \nabla u \cdot \nabla v \quad \text{et} \quad \ell(v) = \int_\Omega fv$$



Éléments finis triangles de type (1)



$$\varphi_i = \lambda_i, \quad 1 \leq i \leq 3$$

où

$$\lambda_i = x_i, \quad 1 \leq i \leq n \quad \text{et } \lambda_{n+1} = 1 - \sum_{i=1}^{n} \lambda_i$$

$$\lambda_1 = x, \quad \lambda_2 = y, \quad \lambda_3 = 1 - x - y$$

Donc

$$\boxed{\varphi_1 = x, \quad \varphi_2 = y, \quad \varphi_3 = 1 - x - y}$$

Matrice élémentaire

- Matrice $A = (a_{ij})_{1 \leq i,j \leq 3}$ où

$$a_{ij} = a(\varphi_i, \varphi_j) = \int_\Omega \nabla\varphi_i \cdot \nabla\varphi_j$$

$$\nabla\varphi_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad \nabla\varphi_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad \nabla\varphi_3 = \begin{pmatrix} -1 \\ -1 \end{pmatrix}$$

$$\boxed{A = \frac{1}{2}\begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}}$$

- Second membre $B = (b_i)_{1 \leq i \leq 3}$ où

$$\boxed{b_i = \ell(\varphi_i) = \int_\Omega f\varphi_i}$$

- Table de nœuds :

n	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
i	0	1	2	3	0	1	2	3	0	1	2	3	0	1	2	3
j	0	0	0	0	1	1	1	1	2	2	2	2	3	3	3	3

- Table de connectivité :

e	0	1	2	3	4	5	6	7	8	9	10	11	12	...	17
0	0	5	1	6	2	7	4	9	5	10	6	11	8	...	15
1	5	0	6	1	7	2	9	4	10	5	11	6	13	...	10
2	4	1	5	2	6	3	8	5	9	6	10	7	12	...	11

Éléments finis 1D

Considérons le problème suivant :

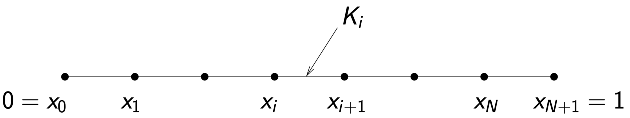
$$(\mathscr{P})\begin{cases} -\dfrac{\mathrm{d}^2u}{dx^2} = p & \text{dans } \Omega =]0, L[\\ u = 0 & \text{sur } \Gamma_1 = \{0\} \\ \dfrac{\mathrm{d}u}{dx} = f & \text{sur } \Gamma_2 = \{L\} \end{cases}$$

Formulation variationnelle : $(\mathscr{P}_v)\begin{cases} \text{Trouver } u \in V \text{ vérifiant} \\ a(u,v) = \ell(v) \quad \forall v \in V \end{cases}$

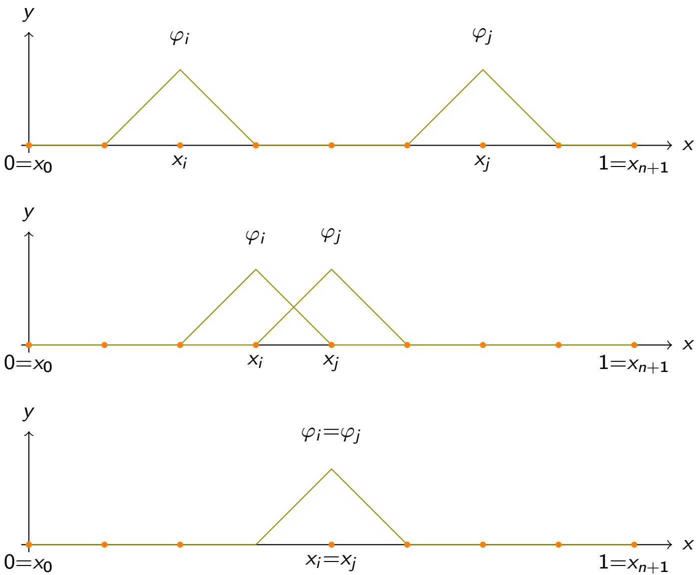
où

$$a(u,v) = \int_0^L \frac{\mathrm{d}u}{dx}\cdot\frac{\mathrm{d}v}{dx}\,\mathrm{d}x \quad \text{et} \quad \ell(v) = \int_0^L p\cdot v(x)\mathrm{d}x + f_L\cdot v(L)$$

Maillage



Eléments finis sans matrice élémentaire

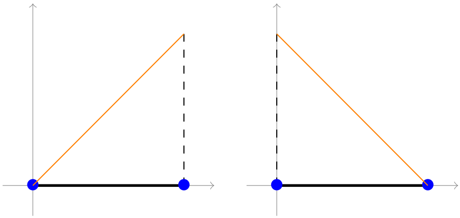


- $\boxed{k=1}$ le segment de type (1) est obtenu pour

$$\mathbb{P} = \mathbb{P}_1^{(1)}, \quad \Sigma = \Sigma_1^{(1)} = \{a_1 = 1, a_2 = 0\}$$

Les fonctions de base sont les fonctions coordonnées barycentriques par rapport à (a_1, a_2), i.e.

$$\varphi_1 = \lambda_1 = \xi, \quad \varphi_2 = \lambda_2 = 1 - \xi$$



$\varphi_1(\xi)=\xi$ $\varphi_2(\xi)=1-\xi$

Construction de l'espace V_h

$$a(\varphi_1,\varphi_1) = a(\varphi_1,\varphi_1) = \frac{1}{h}$$

$$a(\varphi_1,\varphi_2) = a(\varphi_2,\varphi_1) = -\frac{1}{h}$$

Le système élémentaire s'écrit :

$$\begin{pmatrix} f_1 \\ f_2 \end{pmatrix} = \frac{1}{h}\begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}\begin{pmatrix} u_1 \\ u_2 \end{pmatrix}$$

Assemblage en dim=1

On choisit $n = 4$, soit $(u_0, u_1, u_2, u_3, u_4)$ les déplacements aux 5 nœuds. La matrice élémentaire de l'élément e_i, $i = 0, 3$ est donnée par

$$\begin{pmatrix} f_i \\ f_{i+1} \end{pmatrix} = \frac{EA_i}{h_i} \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} u_i \\ u_{i+1} \end{pmatrix}$$

Soit explicitement pour chacune des 4 barres :

$$\begin{pmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \\ f_4 \end{pmatrix} = \begin{pmatrix} 1 & -1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} u_0 \\ u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix}$$

$$\begin{pmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \\ f_4 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} u_0 \\ u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix}$$

Assemblage en dim=1

$$\begin{pmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \\ f_4 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} u_0 \\ u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix}$$

$$\begin{pmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \\ f_4 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & -1 & 1 \end{pmatrix} \begin{pmatrix} u_0 \\ u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix}$$

La matrice d'assemblage s'obtient en sommant les 4 matrices élémentaires :

$$\begin{pmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \\ f_4 \end{pmatrix} = \begin{pmatrix} 1 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & 1 \end{pmatrix} \begin{pmatrix} u_0 = 0 \\ u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix}$$

On $\delta_0 = 0$, La première ligne nous donne la réaction à l'origine $f_0 = -\frac{EA_0}{h_0}\delta_1$. Le système n'a que 4 inconnues, on obtient le système d'ordre 4 en supprimant la première ligne et la première colonne :

$$\begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{pmatrix} = \begin{pmatrix} 2 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 1 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix}$$

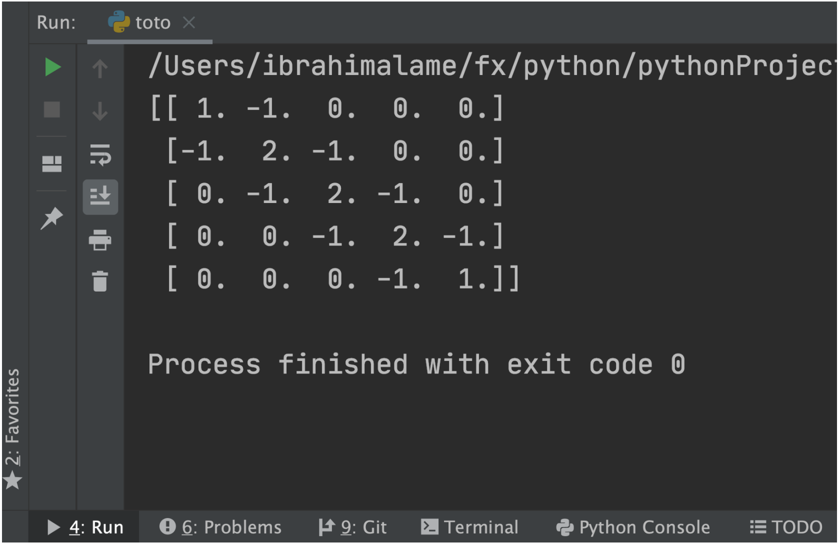
Code python

```python
import numpy as np
N = 5
M = np.zeros((N, N), dtype=float)


def n(e, i):
    if i == 0:
        return e
    elif i == 1:
        return e + 1


for e in range(4):
    K = np.mat([[1, -1], [-1, 1]])
    for ie in range(2):
        i = n(e, ie)
        for je in range(2):
            j = n(e, je)
            M[i, j] += K[ie, je]
print(M)
```

Code python


Run:
toto
/Users/ibrahimalame/fx/python/pythonProjec
[[1. -1. 0. 0. 0.]
 [-1. 2. -1. 0. 0.]
 [0. -1. 2. -1. 0.]
 [0. 0. -1. 2. -1.]
 [0. 0. 0. -1. 1.]]
Process finished with exit code 0
2: Favorites
4: Run
6: Problems
9: Git
Terminal
Python Console
TODO

Code python

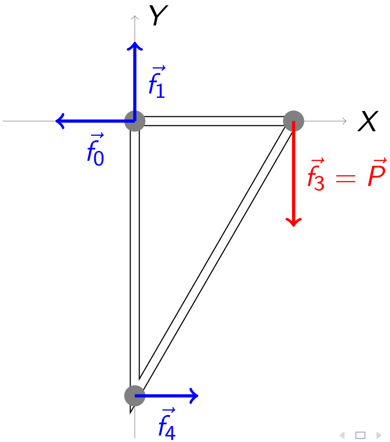
```python
import numpy as np
N = 5
M = np.zeros((N, N), dtype=float)

K = np.mat([[1, -1], [-1, 1]])
for e in range(4):
    M[e, e] += K[0, 0]
    M[e, e+1] += K[0, 1]
    M[e + 1, e ] += K[1, 0]
    M[e+1, e + 1] += K[1, 1]
print(M)
```

Treillis soumis à une force nodale

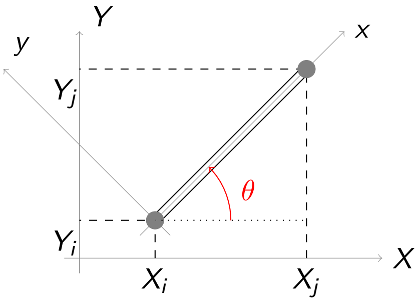
trois poutres de même nature et de même section droite.

- Le noeud 1 est articulé et le nœud 3 repose sur un appui simple dont la normale est horizontale.
- Le noeud 2 porte une charge de composantes (0, P).



Le système élémentaire s'écrit :

$$\begin{pmatrix} f_1 \\ f_2 \end{pmatrix} = \frac{1}{h} \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}$$



Formules de passage

Les formules de passage entre les deux repères local (x, y) et global (X, Y) :

$$\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} \cos\theta & \sin\theta & 0 & 0 \\ 0 & 0 & \cos\theta & \sin\theta \end{pmatrix} \begin{pmatrix} U_1^x \\ U_1^y \\ U_2^x \\ U_2^y \end{pmatrix}$$

$$\begin{pmatrix} F_1^x \\ F_1^y \\ F_2^x \\ F_2^y \end{pmatrix} = \begin{pmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & \cos\theta \\ 0 & \sin\theta \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \end{pmatrix}$$

Matrice de rigidité élémentaire dans le repère global

$$\begin{pmatrix} F_1^x \\ F_1^y \\ F_2^x \\ F_2^y \end{pmatrix} = \frac{1}{L} \begin{pmatrix} C^2 & CS & -C^2 & -CS \\ CS & S^2 & -CS & -S^2 \\ -C^2 & -CS & C^2 & CS \\ -CS & -S^2 & CS & S^2 \end{pmatrix} \begin{pmatrix} U_1^x \\ U_1^y \\ U_2^x \\ U_2^y \end{pmatrix}$$

Changement de notations !

$$\begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{pmatrix} = \frac{EA}{L} \begin{pmatrix} C^2 & CS & -C^2 & -CS \\ CS & S^2 & -CS & -S^2 \\ -C^2 & -CS & C^2 & CS \\ -CS & -S^2 & CS & S^2 \end{pmatrix} \begin{pmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{pmatrix}$$

Les caractéristique géométriques du système se résume en :

nœud	x	y
0	0	0
1	L	0
2	0	$-\sqrt{3}L$

poutre	ℓ	θ	$C = \cos\theta$	$S = \sin\theta$
(0,1)	L	0	1	0
(1,2)	$2L$	$-\frac{2\pi}{3}$	$-1/2$	$-\sqrt{3}/2$
(2,0)	$\sqrt{3}L$	$\frac{\pi}{2}$	0	1

Écrivons l'équation d'équilibre pour chaque barre :

$$\begin{pmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \end{pmatrix} = \frac{EA}{L} \begin{pmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \times \begin{pmatrix} \delta_0 \\ \delta_1 \\ \delta_2 \\ \delta_3 \end{pmatrix}$$

$$\begin{pmatrix} f_2 \\ f_3 \\ f_4 \\ f_5 \end{pmatrix} = \frac{EA}{8L} \begin{pmatrix} 1 & \sqrt{3} & -1 & -\sqrt{3} \\ \sqrt{3} & 3 & -\sqrt{3} & -3 \\ -1 & -\sqrt{3} & 1 & \sqrt{3} \\ -\sqrt{3} & -3 & \sqrt{3} & 3 \end{pmatrix} \times \begin{pmatrix} \delta_2 \\ \delta_3 \\ \delta_4 \\ \delta_5 \end{pmatrix}$$

$$\begin{pmatrix} f_4 \\ f_5 \\ f_0 \\ f_1 \end{pmatrix} = \frac{EA}{\sqrt{3}L} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} \delta_4 \\ \delta_5 \\ \delta_0 \\ \delta_1 \end{pmatrix}$$

Les systèmes élémentaires dans la base globale

Ensuite, nous écrivons les trois systèmes en base 6 en fonction d'une même inconnue $(\delta_0, \delta_1, \delta_2, \delta_3, \delta_4, \delta_5)^t$:

$$\begin{pmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \end{pmatrix} = \frac{EA}{L} \begin{pmatrix} 1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \times \begin{pmatrix} \delta_0 \\ \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \\ \delta_5 \end{pmatrix}$$

$$\begin{pmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \end{pmatrix} = \frac{EA}{L} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{8} & \frac{\sqrt{3}}{8} & -\frac{1}{8} & -\frac{\sqrt{3}}{8} \\ 0 & 0 & \frac{\sqrt{3}}{8} & \frac{3}{8} & -\frac{\sqrt{3}}{8} & -\frac{3}{8} \\ 0 & 0 & -\frac{1}{8} & -\frac{\sqrt{3}}{8} & \frac{1}{8} & \frac{\sqrt{3}}{8} \\ 0 & 0 & -\frac{\sqrt{3}}{8} & -\frac{3}{8} & \frac{\sqrt{3}}{8} & \frac{3}{8} \end{pmatrix} \times \begin{pmatrix} \delta_0 \\ \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \\ \delta_5 \end{pmatrix}$$

Les systèmes élémentaires dans la base globale

$$\begin{pmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \end{pmatrix} = \frac{EA}{L} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{\sqrt{3}} & 0 & 0 & 0 & -\frac{1}{\sqrt{3}} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{1}{\sqrt{3}} & 0 & 0 & 0 & \frac{1}{\sqrt{3}} \end{pmatrix} \times \begin{pmatrix} \delta_0 \\ \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \\ \delta_5 \end{pmatrix}$$

Matrice d'assemblage

$$\begin{pmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \end{pmatrix} = \frac{EA}{L} \begin{pmatrix} 1 & 0 & -1 & 0 & 0 & 0 \\ 0 & \frac{1}{\sqrt{3}} & 0 & 0 & 0 & -\frac{1}{\sqrt{3}} \\ -1 & 0 & 1+\frac{1}{8} & \frac{\sqrt{3}}{8} & -\frac{1}{8} & -\frac{\sqrt{3}}{8} \\ 0 & 0 & \frac{\sqrt{3}}{8} & \frac{3}{8} & -\frac{\sqrt{3}}{8} & -\frac{3}{8} \\ 0 & 0 & -\frac{1}{8} & -\frac{\sqrt{3}}{8} & \frac{1}{8} & \frac{\sqrt{3}}{8} \\ 0 & -\frac{1}{\sqrt{3}} & -\frac{\sqrt{3}}{8} & -\frac{3}{8} & \frac{\sqrt{3}}{8} & \frac{3}{8}+\frac{1}{\sqrt{3}} \end{pmatrix} \times \begin{pmatrix} \delta_0 = 0 \\ \delta_1 = 0 \\ \delta_2 \\ \delta_3 \\ \delta_4 = 0 \\ \delta_5 \end{pmatrix}$$

Les trois déplacements nuls $\delta_0 = \delta_1 = \delta_4 = 0$ nous permettent de réduire le système et de supprimer de la matrice globale les trois lignes 0,1,4 et les trois colonnes 0,1,4. D'où le système linéaire :

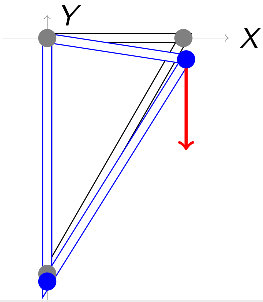
$$\boxed{\begin{pmatrix} 0 \\ P \\ 0 \end{pmatrix} = \frac{EA}{L} \begin{pmatrix} 1+\frac{1}{8} & \frac{\sqrt{3}}{8} & -\frac{\sqrt{3}}{8} \\ \frac{\sqrt{3}}{8} & \frac{3}{8} & -\frac{3}{8} \\ -\frac{\sqrt{3}}{8} & -\frac{3}{8} & \frac{3}{8}+\frac{1}{\sqrt{3}} \end{pmatrix} \times \begin{pmatrix} \delta_2 \\ \delta_3 \\ \delta_5 \end{pmatrix}}$$

Résolution du système

Après avoir résolu le système linéaire réduit, on reprend les équations éliminées du système globale pour déterminer les réactions aux appuis :

$$f_0 = -\frac{EA}{L}\delta_2$$
$$f_1 = -\frac{EA}{L}\frac{1}{\sqrt{3}}\delta_5$$
$$f_4 = \frac{EA}{L}\left(-\frac{\sqrt{3}}{8}\delta_2 - \frac{3}{8}\delta_3 + (\frac{3}{8} + \frac{1}{\sqrt{3}})\delta_5\right)$$

On trouve $\delta_2 \simeq 0.06$mm, $\delta_3 \simeq -0.47$mm, $\delta_5 \simeq -0.17$mm.
et $f_0 \simeq -5.8 \times 10^3$N, $f_1 \simeq 1.0 \times 10^4$N, $f_4 \simeq 5.8 \times 10^3$N.



Géométrie du problème

$$Points = ([x_i, y_i])_{i=0,n-1}$$
$$Barres = ([p_i, p_j])_{(i,j)\in G}$$

En python N et B sont codés par les deux listes suivantes :

```
L = .7
Points = [[0, 0], [0, L], [L, 0]]
Barres = [[0, 1], [1, 2], [2, 0]]
```

Un nœud peut être articulé en un point fixe qui empêche tout déplacement $u_1 = u_2 = 0$, ou un appui simple horizontal ($u_2 = 0$) ou vertical ($u_1 = 0$). On représente la fixation d'un nœud par un triplet (p_i, α_i, β_i) où p_i est le point considéré, α_i est un boolean qui vaut 1 si le déplacement horizontal est libre, 0 sinon, et β_i est un boolean qui vaut 1 si le déplacement vertical est libre, 0 sinon. Pour notre exemple :

```
Conditions = [[0,0,0],[2,0,1]]
```

Les constantes et grandeurs physiques

La section des barres A exprimée en m^2. Les longueurs des barres L_i sont calculées à partir des coordonnées des points d'articulation et exprimées en mètre (m). Le module de Young E est exprimé en Pascal (Pa) . Les forces extérieurs nodales sont exprimées en Newton (N).
Pour notre exemple nous avons : $L = 0.2\mathrm{m}$, $A = 100\mathrm{m}^2$, $E = 200000\mathrm{MPa}$, et $P = -10000\mathrm{N}$.

```
L = 0.2
A = 100 * 1E-6
E = 200000 * 1E6
P = -10000
```

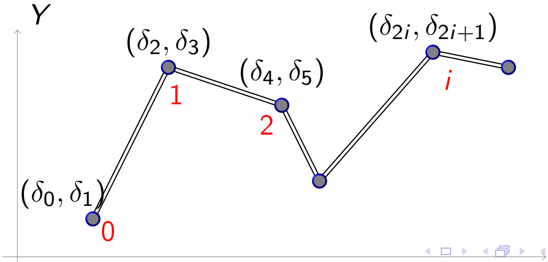
Indexation

Soit $B_i = (p1, p2)$ une barre du système d'extrémités $p_1 = (X_1, Y_1)$ et $p_2 = (X_2, Y_2)$. On calcule ℓ la longueur de la barre, c et s, le cosinus et le sinus de l'angle θ que fait la barre avec l'axe (OX) par :

$$c = \cos\theta = \frac{X_2 - X_1}{\ell} \text{ et } s = \sin\theta = \frac{Y_2 - Y_1}{\ell}, \quad \ell = \sqrt{(X_2 - X_1)^2 + (Y_2 - Y$$

Si on désigne par CS la matrice ligne $\begin{pmatrix} c & s & -c & -s \end{pmatrix}$, la matrice de rigidité n'est autre que

$$\mathsf{K}_i = \frac{EA}{\ell} CS \times CS^t$$



Indexation

Les nœuds sont numérotés $0, 1, 2, \dots, q, \dots, r, \dots n-1$, on désigne par (u_{2q}, u_{2q+1}) le vecteur déplacement du nœud q. L'équation matricielle d'équilibre d'une barre (q, r) s'écrit matriciellement en dimension $2n$:

$$\begin{bmatrix} \vdots \\ \vdots \\ f_{2q} \\ f_{2q+1} \\ \vdots \\ \vdots \\ f_{2r} \\ f_{2r+1} \\ \vdots \end{bmatrix} = \frac{EA}{\ell} \begin{bmatrix} & \vdots & \vdots & & \vdots & \vdots & \\ & \vdots & \vdots & & \vdots & \vdots & \\ \dots & C^2 & CS & \dots & -C^2 & -CS & \dots \\ \dots & CS & S^2 & \dots & -CS & -S^2 & \dots \\ & \vdots & \vdots & & \vdots & \vdots & \\ & \vdots & \vdots & & \vdots & \vdots & \\ \dots & -C^2 & -CS & \dots & C^2 & CS & \dots \\ \dots & -CS & -S^2 & \dots & CS & S^2 & \dots \\ & \vdots & \vdots & & \vdots & \vdots & \end{bmatrix} \times \begin{bmatrix} \vdots \\ \vdots \\ \delta_{2q} \\ \delta_{2q+1} \\ \vdots \\ \vdots \\ \delta_{2r} \\ \delta_{2r+1} \\ \vdots \end{bmatrix}$$

Les autres coefficients manquant sont tous nuls.

Matrice d'assemblage

```
for p1, p2 in Barres:
    x1 = Points[p1][0]
    y1 = Points[p1][1]
    x2 = Points[p2][0]
    y2 = Points[p2][1]

    ell = math.sqrt((x2 - x1) ** 2 + (y2 - y1) ** 2)
    c = (x2 - x1) / ell
    s = (y2 - y1) / ell
    CS = np.mat([c, s, -c, -s], dtype=float)
    CSt = np.transpose(CS)
    m = np.dot(CSt, CS)*A*E/ell

    M[2 * p1, 2 * p1] += m[0, 0]
    M[2 * p1, 2 * p1 + 1] += m[0, 1]
```

Matrice d'assemblage

```
M[2 * p1, 2 * p2] += m[0, 2]
M[2 * p1, 2 * p2 + 1] += m[0, 3]
M[2 * p1 + 1, 2 * p1] += m[1, 0]
M[2 * p1 + 1, 2 * p1 + 1] += m[1, 1]
M[2 * p1 + 1, 2 * p2] += m[1, 2]
M[2 * p1 + 1, 2 * p2 + 1] += m[1, 3]
M[2 * p2, 2 * p1] += m[2, 0]
M[2 * p2, 2 * p1 + 1] += m[2, 1]
M[2 * p2, 2 * p2] += m[2, 2]
M[2 * p2, 2 * p2 + 1] += m[2, 3]
M[2 * p2 + 1, 2 * p1] += m[3, 0]
M[2 * p2 + 1, 2 * p1 + 1] += m[3, 1]
M[2 * p2 + 1, 2 * p2] += m[3, 2]
M[2 * p2 + 1, 2 * p2 + 1] += m[3, 3]
```

Matrice d'assemblage

Les conditions aux appuis impose un ou deux déplacements nuls. On retire donc du système matriciel les lignes et colonnes correspondants, soit en python :

```
Conditions = [[0, 0, 0], [2, 0, 1]]
l = []
for q, a, b in Conditions:
    if a == 0:
        l.append(2 * q)
    if b == 0:
        l.append(2 * q + 1)

l.sort()
l.reverse()
for i in l:
    M = np.delete(M, i, axis=0)
    M = np.delete(M, i, axis=1)
```

Exemple