C++ Projet

求解公式：线性方程（gradient conjugué 求解）

边界 $u\in R^{I}$, $\forall i\in I_\Gamma$, $u_i = g_i$

$\forall v \in R^{I_0}$ $\sum_k (DN^k(v)\cdot DN^k(u))/\|N^k(w)\| = -\sum_k (DN^k(v)\cdot N^k(0))/\|N^k(w)\|$

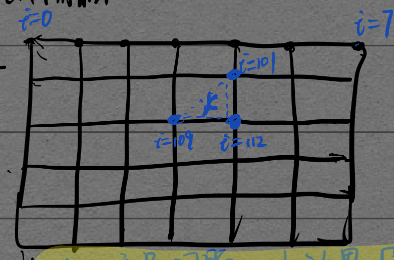
任意　donnée　donnée

$DN^k(u) = \sum_{i=0}^{2} u_g^i (p^{i+2}-p^{i+1})^\perp$, $\binom{i}{g}$ est le numéro du sommet global du sommet i de k

P是二维的 (x,y)

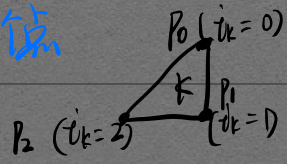
$= u_g^0(p^2-p^1)^\perp + u_g^1(p^3-p^2)^\perp + u_g^2(p^4-p^3)^\perp$　C++下标索引从0开始

$= u_g^0(p^2-p^1)^\perp + u_g^1(p^0-p^2)^\perp + u_g^2(p^1-p^0)^\perp$



i的序号问题：从边界Γ开始定号

总之，u_g^0 与 k 的另外两点



A: vertex

$N^k = (A^j - A^i)\wedge(A^k - A^i)$ 三维　　$N^k(0) = [0,0,\det(p^1-p^0, p^2-p^0)]$

$DN^k(u) = N^k(u) - N^k(0)$　（Z 坐标相同，相减为零．所以 $DN^k(u)$ 是二维的．

定值　定值　定值

$\sum_k (DN^k(v)\cdot DN^k(u))/\|N^k(w)\| = \left(-\sum_k (DN^k(v)\cdot N^k(0))/\|N^k(w)\|\right) = 0$　式1

首先根据给定的 w，求出 u．然后 $w = u^i$，求 u^{i+1}．迭代求解最终结果．

(l'algorithme du point fixe)

对于 $v_i \in R^{I_0}$，这个式子最终得到的是 $\sum_{j=1}^{\tilde N} a_{ij}u_j - b_i = 0$

$i = 1,2,\dots N_{I_0} = N_I + N_{\Gamma_I}$ $\Rightarrow$ $A_{ij}\,u = b$ un système linéaire avec gradient conjugué

$j = 1,2,3,\dots, N = N_I$

进一步的推导

验证两种方式得到的 $DN^k(u)$ 是一致

$$\underline{N^k = (A^j - A^i)\wedge(A^k - A^i)} = \begin{cases} -(y_j-y_i)\times(z_k-z_i) + (z_j-z_i)(y_k-y_i)\cdot i \\ -(z_j-z_i)(x_k-x_i) + (x_j-x_i)(z_k-z_i)\cdot j \\ -(x_j-x_i)(y_k-y_i) + (y_j-y_i)(x_k-x_i)\cdot k \end{cases}$$

$$= y_jz_k - y_jz_i - y_iz_k + \cancel{y_iz_i} - z_jy_k + z_jy_i + z_iy_k - \cancel{z_iy_i}$$

$$z_jx_k - z_jx_i - z_ix_k + \cancel{z_ix_i} - x_jz_k + x_jz_i + x_iz_k - \cancel{x_iz_i}$$

$$= \begin{cases} z_i(y_k - y_j) + z_j(y_i - y_k) + z_k(y_j - y_i) \\ z_i(x_j - x_k) + z_j(x_k - x_i) + z_k(x_i - x_j) \end{cases}$$

$$DN^k(u) = u_g^0 (p^2 - p^1)^\perp + u_g^1 (p^0 - p^2)^\perp + u_g^2 (p^1 - p^0)^\perp$$

$p^0(x_i, y_i)$, $p^1(x_j, y_j)$, $p^2(x_k, y_k)$, $u_g^0 = z_i$, $u_g^1 = z_j$, $u_g^2 = z_k$.

$$p^2 - p^1 = \begin{bmatrix} x_k - x_j \\ y_k - y_j \end{bmatrix}, \quad p^0 - p^2 = \begin{bmatrix} x_i - x_k \\ y_i - y_k \end{bmatrix}, \quad p^1 - p^0 = \begin{bmatrix} x_j - x_i \\ y_j - y_i \end{bmatrix}$$

$$DN^k(u) = z_i \begin{pmatrix} y_j - y_k \\ x_k - x_j \end{pmatrix} + z_j \begin{bmatrix} y_k - y_i \\ x_i - x_k \end{bmatrix} + z_k \begin{bmatrix} y_i - y_j \\ x_j - x_i \end{bmatrix}$$

$$N^k(u) = [0, 0, \det(p^1 - p^0, p^2 - p^0)]$$

$$\det(p^1 - p^0, p^2 - p^0) = \begin{vmatrix} x_j - x_i & x_k - x_i \\ y_j - y_i & y_k - y_i \end{vmatrix}$$

代码

编写函数：DN^k, $\text{vector}^\perp$, N^k, $\|N^k\|$

① $\sum_k (DN^k(v) \cdot DN^k(u)) / \|N^k(w)\| = -\sum_k (DN^k(v) \cdot N^k(v)) / \|N^k(w)\|$ $= 0$

$$DN^k(v) = \begin{cases} DNv_x \\ DNv_y \end{cases}$$ 可以用来求面积

$$\begin{cases} a_i \Leftarrow \dfrac{Dv_x(y_k - y_j) + Dv_y(x_j - x_k)}{\|N^k(w)\|} \cdot u_i \\ a_j \Leftarrow \dfrac{Dv_x(y_i - y_k) + Dv_y(x_k - x_i)}{\|N^k(w)\|} \cdot u_j \\ a_k \Leftarrow \dfrac{Dv_x(y_j - y_i) + Dv_y(x_i - x_j)}{\|N^k(w)\|} \cdot u_2 \end{cases}$$

② $\sum_k \dfrac{[DN^k(v) \cdot DN^k(u)]}{\|N^k(w)\|} = 0$, $u \in R^I$, $\forall i \in I_T$, $u_i = g_i$, $w \in R^I$ soit aussi donné

$$DN^k(v) = v_g^0 (p^2 - p^1)^\perp + v_g^1 (p^0 - p^2)^\perp + v_g^2 (p^1 - p^0)^\perp$$

$$DN^k(u) = \begin{bmatrix} \times & \times & \times \\ \times & \times & \times \\ 0 & 0 & 0 \end{bmatrix} u \begin{bmatrix} u_0 \\ u_1 \\ u_2 \end{bmatrix}, \quad DN^k(v) = \underbrace{\begin{bmatrix} \times & \times & \times \\ \times & \times & \times \\ 0 & 0 & 0 \end{bmatrix}}_{A_i} v \begin{bmatrix} v_0 \\ v_1 \\ v_2 \end{bmatrix}$$

A_i

只有 v_0, v_1, v_2 中为1 对应的全局点有机会

vecteur de base $[1 \cdots\cdots 0]$

A_i是完全相同的

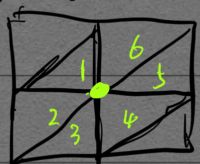
$$DN^k(u) = z_i\begin{pmatrix} y_j - y_k \\ x_k - x_j\end{pmatrix} + z_j\begin{bmatrix} y_k - y_i \\ x_i - x_k\end{bmatrix} + z_k\begin{bmatrix} y_i - y_j \\ x_j - x_i\end{bmatrix}, \quad i = P_0,\ j = P_1,\ k = P_2$$

$$= \underset{u_0}{P_0.z}\begin{bmatrix} P_1.y - P_2.y \\ P_2.x - P_1.x\end{bmatrix} + \underset{u_1}{P_1.z}\begin{bmatrix} P_2.y - P_0.y \\ P_0.x - P_2.x\end{bmatrix} + \underset{u_2}{P_2.z}\begin{bmatrix} P_0.y - P_1.y \\ P_1.x - P_0.x\end{bmatrix}$$

③

$$\begin{bmatrix} DNvx \\ DNvy \end{bmatrix} \quad DN^k(u) = \begin{bmatrix} P_1.y - P_2.y & P_2.y - P_0.y & P_0.y - P_1.y \\ P_2.x - P_1.x & P_0.x - P_2.x & P_1.x - P_0.x \\ 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} u_0 \\ u_1 \\ u_2 \end{bmatrix} \begin{matrix} \to P_0.Z \\ \to P_1.Z \\ \to P_2.Z \end{matrix}$$

每一项都不为0



无论如何，每一个点都对应一个 global index，所以只有与这个点相关的系数才为0.

$$DN^k(u) = \overset{A_i}{\begin{bmatrix} \times & \times & \times \\ \times & \times & \times \\ 0 & 0 & 0 \end{bmatrix}}\begin{bmatrix} u_0 \\ u_1 \\ u_2 \end{bmatrix}, \qquad DN^k(v) = \overset{A_i}{\begin{bmatrix} \times & \times & \times \\ \times & \times & \times \\ 0 & 0 & 0 \end{bmatrix}}\begin{bmatrix} v_0 \\ v_1 \\ v_2 \end{bmatrix}$$

$$DN^k(v) \cdot DN^k(u) = v^T \underline{A_i^T A_i}\, u = [v_0\ v_1\ v_2]\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}\begin{bmatrix} u_0 \\ u_1 \\ u_2 \end{bmatrix}$$

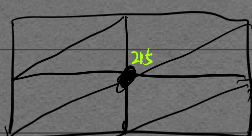
若 v_0, v_1, v_2 均为0，点积为0，所有 u_0 系数为0.

若 $v_i = 1$，其他为0. $[v_0\ v_1\ v_2]\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = [a_{i1}\ a_{i2}\ a_{i3}]$

$$[a_{i1}\ a_{i2}\ a_{i3}]\begin{bmatrix} u_0 \\ u_1 \\ u_2 \end{bmatrix} = a_{i1}u_0 + a_{i2}u_1 + a_{i3}u_2$$

所以只与 u_i 相连的点的系数才不为0. i 是 v_i 为1的那个点 global index

并且所有系数 a_{ij} 是固定的，当网格确定



Produit scalaire

④

$$DNvx\left[(P_1.y - P_2.y)u_0 + (P_2.y - P_0.y)u_1 + (P_0.y - P_1.y)u_2\right]$$

$$+ DNvy\left[(P_2.x - P_1.x)u_0 + (P_0.x - P_2.x)u_1 + (P_1.x - P_0.x)u_2\right]$$

$$a_0 \Leftarrow \left[DNvx(P_1.y - P_2.y) + DNvy(P_2.x - P_1.x)\right]u_0 +$$

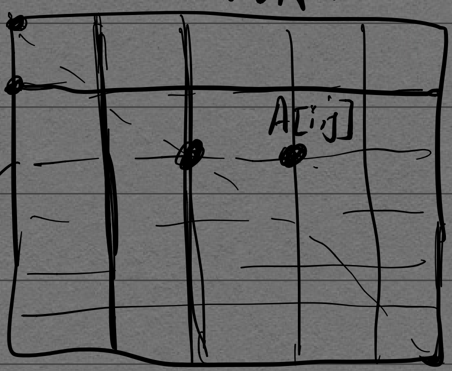
$$a_1 \Leftarrow \left[DNvx(P_2.y - P_0.y) + DNvy(P_0.x - P_2.x)\right]u_1 +$$

$a_2 \leftarrow [DNUx\,(P_0.y - P_1.y) + DNUy\,(P_1.x - P_0.x)]\; u_2$

MatA : N×N

行

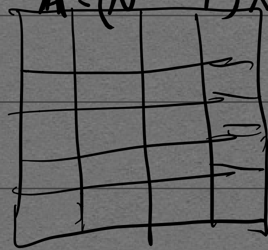
5



$\begin{bmatrix} u_0 \\ u_1 \\ u_2 \\ \vdots \\ u_{N-1} \end{bmatrix} = 0 \Rightarrow$

$AU_{/\Gamma} = b$ seconde membre

$A: (N-N_\Gamma)\times(N-N_\Gamma)$



$\begin{bmatrix} u_{/\Gamma} \end{bmatrix} = \begin{bmatrix} b \end{bmatrix}$

$b_i = \sum_{j=0}^{N-1} -u[j] * MatA[i,j]$

对每一个非边界点的行进行搜索，若发现边界点，移动到b中.

AU=b   Decomp MatA   OK

老师代码分析 I：EF2d-base.hpp

(1) R2.hpp   R2 是一个二维坐标的点   R2(x,y)

+=, -=, +, -, (,) produit scalaire, ^ produit mixte,

* / ⇒ 实数   - : 取负   + : 不变   perp() = (-y, x) la perpendiculaire

( ), [ ] = $\begin{cases} i=0, \to x \\ i=1, \to y \end{cases}$ norme : sqrt(x^2+y^2),

$\det(A,B,C) = R2(A,B) \wedge R2(A,C) = \begin{bmatrix} B_x - A_x \\ B_y - A_y \end{bmatrix} \wedge \begin{bmatrix} C_x - A_x \\ C_y - A_y \end{bmatrix}$

$= (B-A)\wedge(C-A) = (A_x - B_x)(A_y - C_y) - (A_y - B_y)(A_x - C_x)$

(2) EF2d-base.hpp

Label : int lab

Ver : Vertex() { } R_2 + $\boxed{\text{Label}}$ R_2点的拓展

Simplex : nbv = 3 (nombre de vecteur) 三角形   double mes 面积

V[nbv] → V中3个点形成三角形 (初始为0)

t[k].$\boxed{\text{build}(v, I, -1)}$ → offset 根据 maillage 的起始序号，最后要从0开始

v_0：点的坐标，　I：点的序号

$$V[i] = V_0 + I[i] + offset;$$

$mes = det(*V[0], *V[1], *V[2]) * 0.5;$ 获得三角形的面积

$\boxed{V[0]} = \boxed{V_0 + I[0] + offset}$　　$I = [9, 8, 11, 0]$

指针　-1 对地址进行计算，实际上就是索引偏移

$$= V_0[I[0] + offset]$$

Mesh2d (k, i)：

Vertex *v; 保存了所有点的坐标（排序是按照 $0, 1, 2, 3, \cdots nv-1$ 进行排列）

Simplex *t; simplex 数组，保存了所有 simplex 的信息 $(0, 1, 2, 3, \cdots, nt-1)$

Mesh2d $(k, i) \longrightarrow$ Mesh2d $(t[k].v[i])$

$\longrightarrow$ 返回第 k 个三角形的第 i 个顶点的序号

Mesh2d Th; 二维网格

第 k 个三角形：Th[k] (simplex)
$$\begin{cases} \overset{(vertex)}{Th[k][i]} \rightarrow \text{三角形 k 的 i 顶点坐标} \quad i = 0, 1, 2 \cdots \\ Th(\overset{(int)}{Th[k][i]}) \rightarrow \text{返回 i 顶点的序号} \end{cases}$$

void AUB (Th, v, w, A, b)

$$\sum_k (DN^k(v) \cdot DN^k(u)) / \|N^k(w)\| = -\sum_k (DN^k(v) \cdot N^k(w)) / \|N^k(w)\|$$

编写函数：DN^k, $vector^{\perp}$, N^k, $\|N^k\|$

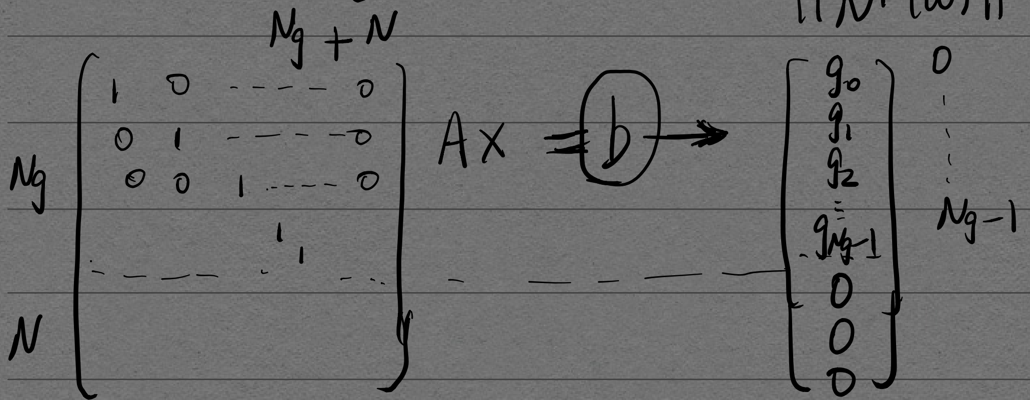
$$\sum_k (DN^k(v) \cdot DN^k(u)) / \|N^k(w)\| = -\sum_k (DN^k(v) \cdot N^k(w)) / \|N^k(w)\|$$

$$DN^k(v) = \begin{cases} DNvx \\ DNvy \end{cases}$$

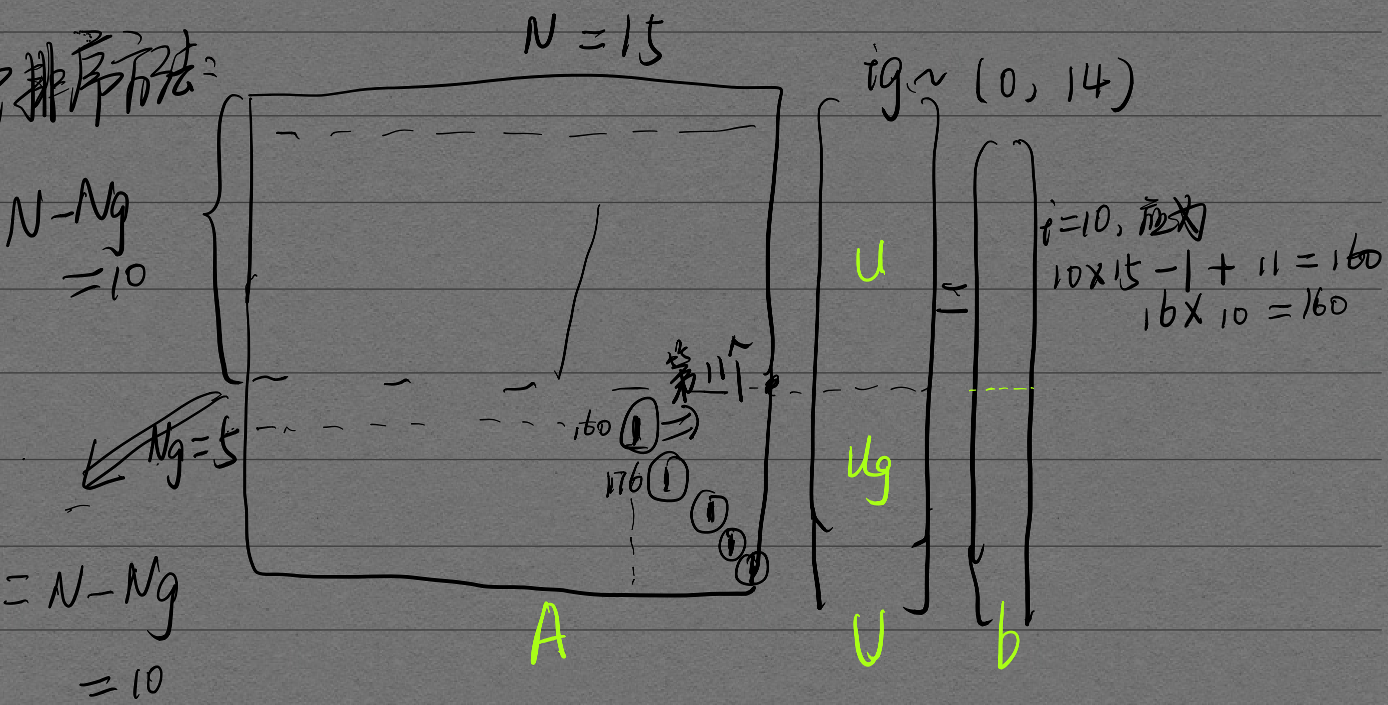
$$\begin{cases} a_i \Leftarrow \dfrac{Dux(y_k - y_j) + Duy(x_j - x_k)}{\|N^k(w)\|} . \quad u_i \rightarrow P_0 \\ a_j \Leftarrow \dfrac{Dux(y_i - y_k) + Duy(x_k - x_i)}{\|N^k(w)\|} \quad u_j \rightarrow P_1 \\ a_k \Leftarrow \dfrac{Dux(y_j - y_i) + Duy(x_i - x_j)}{\|N^k(w)\|} . \quad u_2 \rightarrow P_2 \end{cases}$$

$\|N^k(w)\|$

$$\begin{cases} a_0 = \dfrac{DNKx\,(P_1.y - P_2.y) + DNKy\,(P_2.x - P_1.x)}{\|N^k(w)\|} \cdot u_i \\[2ex] a_1 = \dfrac{DNKx\,(P_2.y - P_1.y) + DNKy\,(P_0.x - P_2.x)}{\|N^k(w)\|} \cdot u_j \\[2ex] a_2 = \dfrac{DNKx\,(P_0.y - P_1.y) + DNKy\,(P_1.x - P_0.x)}{\|N^k(w)\|} \cdot u_2 \end{cases}$$



第一种排序方法：



$i = N - Ng$

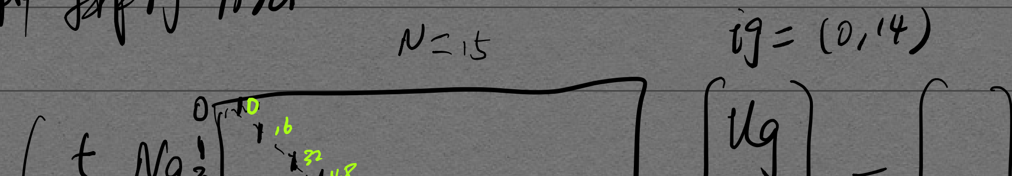
$= 10$

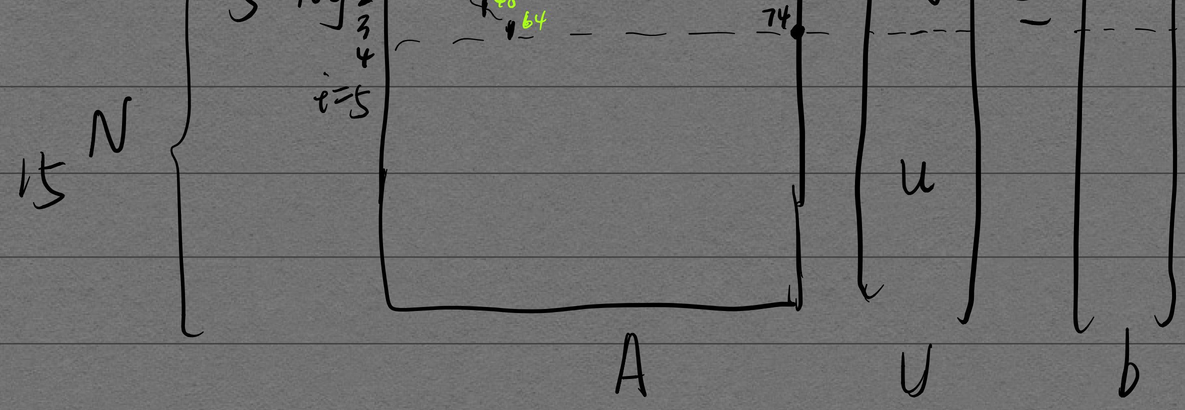
$(Th.nv - Ng) * Th.nv \cancel{-1} + \cancel{i+1} + Th.nv * (i - (Th.nv - Ng))$

$= (Th.nv - Ng + i - Th.nv - Ng) \cdot Th.nv + i$

$= i \times Th.nv + i$

第二种排序方法：



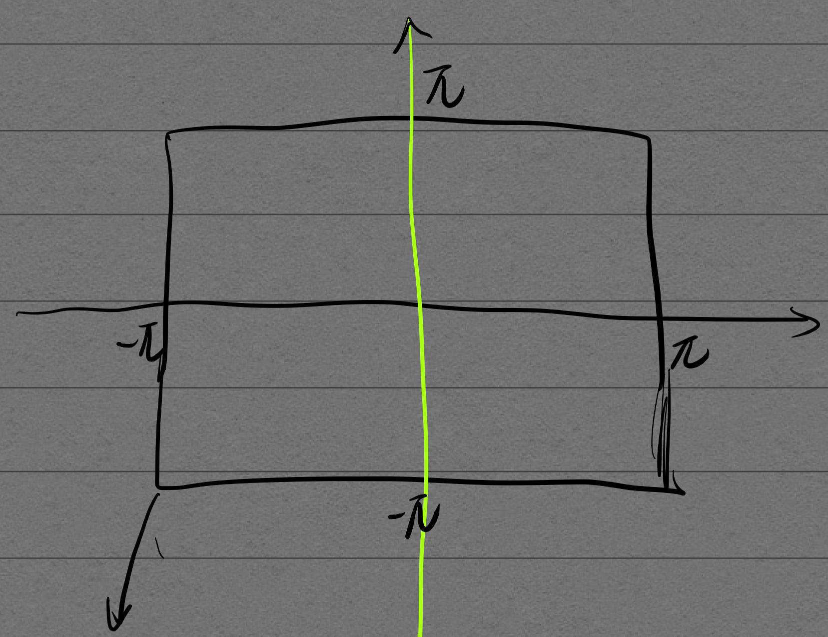


① 数组的初始化

② V的构造函数　　build_V

$V \rightarrow$ matrix $(N-N_g) \times N$

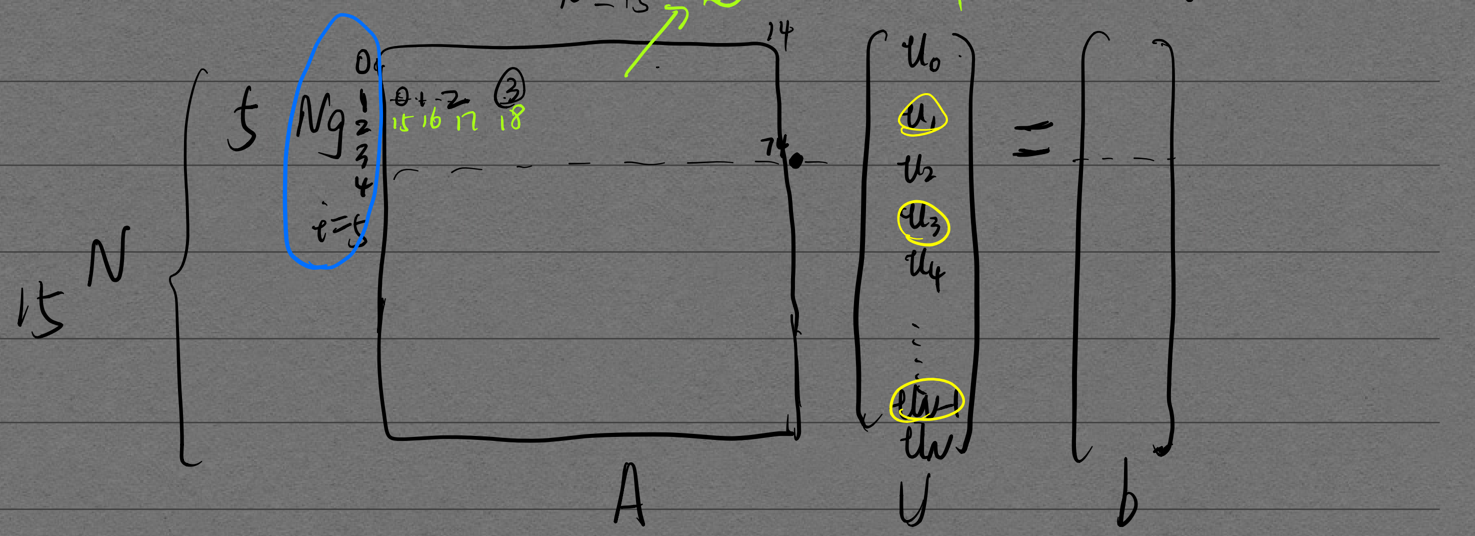
fonction : AUb → calculer la matrix A et b



创建一个序列对应上去，将序列重新排序

这种情况下，g，w的长度都应为 N

$tg = (0, \cdots, 14)$



for (int i=0; i<Th.nv; i++)

if (i<Ng) {

func: borde_idx

j= borde_idx(LocalIdx, i) 提真实的点编号 $j \in [0, N-1]$

b[j] = w[j];

Mat A[i × N ~~-1~~ + j] = 1;

func: borde_idx 返回边界点的全局索引

ConvertLocalIdx 输入Th对象，改变 LocalIdx 存储边界点的全局索引

需要初始化的数组：

g (边界)，W (初始U值，满足边界条件)，
长度为N，需要修改

Mat A (保存系数的数组)，b (Ax=b)，

vector<vector<double>> V：随机 N×N 的矩阵，用来求解系数.

b 既是 Ax=b，也是边界

vector<int> LocalIdx $i \longleftrightarrow j$

1:Ng   对应的边界点索引

调试代码：

ConvertLocalIdx: OK，   build_V OK

maillage: $\Omega =]-\pi, \pi[$, $f(x,y) = \cos(x) \cdot \cos(y)$

build_9 : OK

AVb : OK

106 : NKW OK 107 : norme_NKW OK

老师代码分析2: GC.cpp

GradientConjugue (MatVirt &A, MatVirt &C,
OK double *b second membre, double *x solution qui contient une initialisation, int nbitermax,
double eps, int niveauimpression)

$v = \vec{e_i}$, vecteur de base $v \in \mathbb{R}^{I_0}$

L'algorithme du point fixe en faisant la boucle suivante
$w = u^i$ et calculer u^{i+1} solution Pa.

哪些是需要重新初始化的?

需要: MatA 系数矩阵, A, b (Ax = b)

不需要: W (迭代用 u 替换), V (赋值一次), g 边界值)

Semi-caténoïde

外面为 1, 里面为 2.

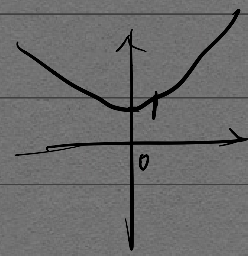
Caténoid 悬链曲面

$$\begin{cases} x = c\cosh\frac{v}{c}\cos u \\ y = c\cosh\frac{v}{c}\sin u \end{cases} \qquad z = v$$

$$x^2+y^2=\left(c\cosh\frac{v}{c}\right)^2$$

$$\sqrt{x^2+y^2}=c\cosh\frac{v}{c}$$

cosh 双曲函数 $\cosh x=\frac{e^x+e^{-x}}{2}$ $[1,+\infty)$



$C=\frac{1}{2}$, $2\sqrt{x^2+y^2}=\cosh 2v$