

THESE

présentée à
l'UNIVERSITE PIERRE ET MARIE CURIE

pour obtenir le titre de

DOCTEUR DE L'UNIVERSITE PARIS 6
spécialité **Mathématiques Appliquées**

par **François DUBOIS**

Sujet de la Thèse :

**QUELQUES PROBLEMES LIES AU CALCUL
D'ECOULEMENTS DE FLUIDES PARFAITS
DANS LES TUYERES.**

Soutenue le 5 janvier 1988 devant le jury composé de

MM.	P.A. RAVIART	Président
	A.Y. LEROUX	
	O. PIRONNEAU	Rapporteurs
	B. MERCIER	
	P. MORICE	
	J.C. NEDELEC	Examineurs

Chapitre 3

Représentation de champs de vecteurs solénoïdaux
de \mathbb{R}^3 par un potentiel vecteur discret :
Résolution numérique d' un problème non-linéaire .

I - GENERALITES

Nous nous proposons dans ce chapitre de décrire avec précision la mise en oeuvre numérique de l'élément fini prismique conforme dans $H(\text{rot})$ introduit par NEDELEC [19,20], pour l'étude d'un écoulement compressible dans une tuyère. Ce travail a été présenté dans [5] en commun avec J.M. Dupuy, qui s'intéressait au problème extérieur (aile NACA 12) à l'aide de l'élément fini tétraédrique de degré 1 (voir aussi DUPUY [8]).

Un fluide parfait compressible stationnaire en évolution isentropique et isenthalpique est complètement décrit par son champ de vitesses \vec{u} et la densité ρ qui vérifient (cf. Chapitre 1) :

$$(1.1) \quad \text{div } \rho \vec{u} = 0 \quad \Omega$$

$$(1.2) \quad \text{rot } \vec{u} = 0 \quad \Omega$$

$$(1.3) \quad \rho^{\gamma-1} = 1 + \frac{\gamma-1}{2} M_0^2 (1 - |\vec{u}|^2) \quad \Omega$$

pour un nombre de Mach de référence M_0 , et un rapport de chaleurs spécifiques γ ($\gamma=1,4$ dans les calculs). Nous avons choisi des conditions aux limites du type flux de masse imposé sur la frontière Γ de Ω .

$$(1.4) \quad \rho \vec{u} \cdot \vec{n} = g \quad \text{sur } \Gamma$$

On décompose la frontière Γ en trois sous ensembles disjoints $\Gamma_0, \Gamma_e, \Gamma_s$ où le flux g est respectivement nul, négatif, positif (figure 1) :

$\Gamma = \Gamma_0 \cup \Gamma_{\text{entrée}} \cup \Gamma_{\text{sortie}}$. Nous avons pris comme valeur numérique :

$$(1.5) \quad g \equiv -1 \quad \text{sur } \Gamma_e, \quad g \equiv 1 \quad \text{sur } \Gamma_s.$$

Si l'écoulement à l'entrée et à la sortie est normal à la frontière, le nombre de Mach peut alors valoir exactement M_0 , mais pas obligatoirement ; lorsque $M_0 < 1$ pour fixer les idées, (1.4) et (1.5) autorisent l'existence d'écoulements supersoniques (cf. NECAS [18] par exemple).

Le flux de masse est représenté par un potentiel vecteur $\vec{\psi} : \rho \vec{u} = \text{rot} \vec{\psi}$ qui est discrétisé dans un espace vectoriel de la forme $Y_h(\Omega) \oplus K_h(\Omega)$, définis au Chapitre 2 (Définitions 5.6 et 4.5 respectivement).

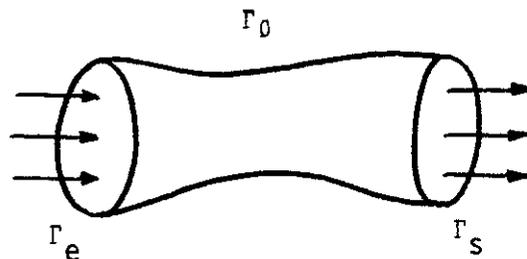


Figure 1

La densité ρ est choisie constante dans chaque élément fini. Nous avons mis en oeuvre un algorithme de point fixe, qui converge toujours si la solution est partout subsonique (GELDER [10], CIAVALDINI [3]).

ALGORITHME

(i) Initialisation

- Calcul de l'unique vecteur $\bar{\psi} \in Y_h(\Omega)$ tel que

$$(1.6) \quad \text{rot}_{\Gamma} \Pi \bar{\psi} = g \quad \text{sur } \Gamma.$$

Le problème (1.6) est aussi appelé "problème de Laplace-Beltrami" dans la suite.

- $\rho^{(0)} \equiv 1$.

(ii) Itération ; $\rho^{(n)}$ est supposé connu.

- Résolution du problème discret

$$(1.7) \quad \begin{cases} \vec{\psi}^{(n+1)} \in K_h(\Omega) \\ \int_{\Omega} \frac{1}{\rho^{(n)}} \text{rot } \vec{\psi}^{(n+1)} \cdot \text{rot } \vec{\varphi} \, dx = - \int_{\Omega} \frac{1}{\rho^{(n)}} \text{rot } \vec{\psi} \cdot \text{rot } \vec{\varphi} \, dx, \forall \vec{\varphi} \in K_h \end{cases}$$

- Calcul d'une moyenne du champ de vitesse dans chaque élément fini K :

$$|\vec{u}_K^{(n+1)}|^2 = \frac{\int_K |\text{rot}(\vec{\psi}^{(n+1)} + \vec{\psi})|^2 \, dx}{\int_K \rho^{(n)} \, dx}$$

- Calcul de la densité $\rho^{(n+1)}$ dans l'élément K grâce au Théorème de Bernoulli :

$$\rho_K^{(n+1)} = \left\{ 1 + \frac{\gamma-1}{2} M_0^2 (1 - |\vec{u}_K^{(n+1)}|^2) \right\}^{\frac{1}{\gamma-1}}$$

(iii) Test de l'erreur $\varepsilon_{n+1} \equiv \left\| \frac{\rho^{(n+1)} - \rho^{(n)}}{\rho^{(n)}} \right\|_{L^\infty(\Omega)}$

- $\varepsilon_{n+1} \geq \varepsilon$: retour en (ii)
- $\varepsilon_{n+1} < \varepsilon$: fin de calcul.

II - DESCRIPTION DU MAILLAGE

1) GENERATION DES SOMMETS

Nous nous sommes bornés à un test simple qui consiste à recalculer à l'aide d'un code tridimensionnel un cas de calcul purement bidimensionnel. Nous avons choisi le "canal du GAMM" proposé par RIZZI et VIVIAND [22] : un profil circulaire d'épaisseur relative 0,42 est placé dans une veine de hauteur OA égale à 2,073 fois la longueur du profil. Le maillage est formé de quadrangles, comporte 72 points le long de Oz et 21 le long de OA. De plus, tous ses sommets sont alignés le long de parallèles à la droite OC (figure 2). Nous donnons à ce canal une épaisseur Y dans la direction Oy et nous maillons la section d'entrée $\Gamma_e \equiv OADE$ du plan xOy à l'aide de triangles. Puis nous engendrons une famille de points dans chaque plan parallèle à la face d'entrée, associé à une ligne $z = cste$ du maillage bidimensionnel initial, par une affinité selon Ox de la face Γ_e . De cette façon, nous avons construit un maillage tridimensionnel de l'intérieur de OABCDEFG par des prismes (figure 3). Comme nous avons choisi un maillage de la face Γ_e régulier (bien que triangulaire), on retrouve dans chaque plan $y = cste$ une trace du maillage quadrangulaire proposé au GAMM Workshop [22]. Le choix des triangles pour le maillage de OADE, donc des prismes pour le maillage tridimensionnel, a été motivé par la possibilité de mailler facilement des figures de révolution autour de Oz, en remplaçant le rectangle OADE par un cercle de centre O, qu'on peut alors mailler par une triangulation non structurée.

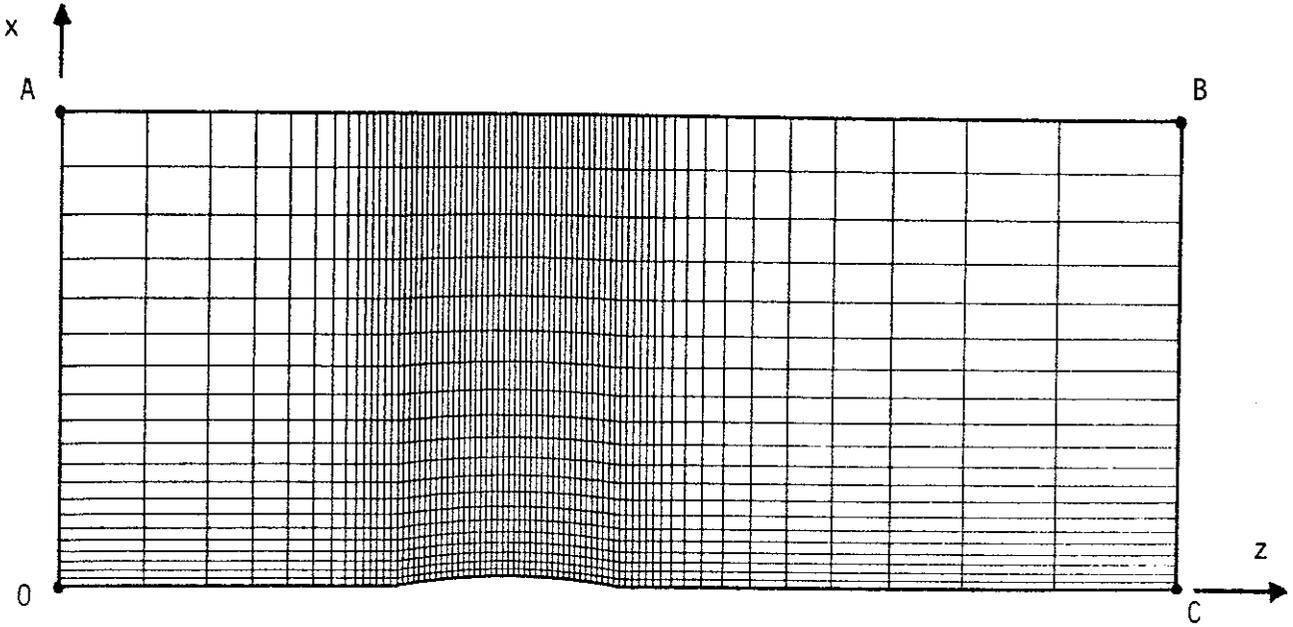


Figure 2 Maillage du canal du GAMM

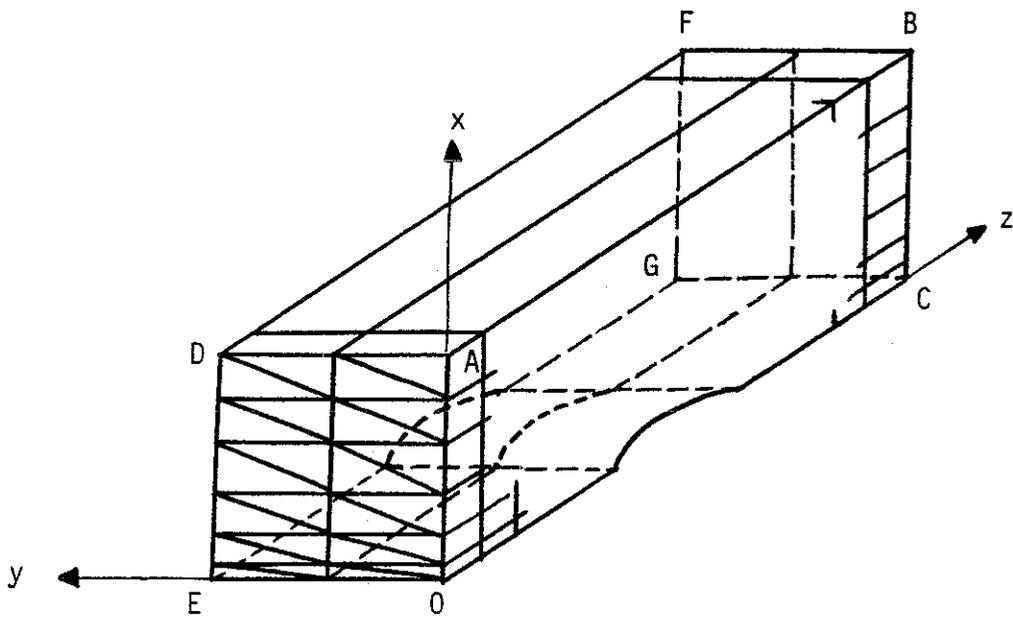


Figure 3

Sur un plan purement informatique, l'aspect déductif de la structure selon Oz permet de ne stocker en mémoire que les informations de topologie du maillage triangulaire ; ses arêtes permettent de repérer l'ensemble des arêtes des plans $z = \text{cste}$, et ses sommets sont associés aux arêtes (presque) parallèles à l'axe Oz. Pour la génération des degrés de liberté, nous avons mis en oeuvre le logiciel MEFISTO [21] (développé à l'Université Paris 6) et plus précisément l'interpolation TRIA MT 10. De cette façon, seuls des fichiers de petite taille sont consacrés au stockage de la topologie semi-déductive du maillage tridimensionnel.

2) MISE EN OEUVRE DE LA CONDITION DE JAUGE

Comme nous l'avons mis en évidence au Chapitre 2, les espaces d'approximation $Y_h(\Omega)$ et $K_h(\Omega)$, du potentiel vecteur sont liés au choix d'un arbre surfacique $T_h(\Gamma)$ et d'un "arbre" intérieur $T_h(\Omega)$ parmi les arêtes du maillage. Nous rappelons que pour une arête a appartenant à $T_h(\Gamma)$ ou à $T_h(\Omega)$, le degré de liberté associé σ_a ,

$$(2.1) \quad \sigma_a(\vec{\psi}) = \int_a \vec{\psi} \cdot d\vec{l}$$

est nul a priori.

La construction de l'arbre $T_h(\Gamma)$ sur la frontière est intimement reliée à la condition limite (1.4), comme l'a remarqué F.X. ROUX [23]: la fonction g est identiquement nulle sur toute la partie Γ_0 de la frontière, donc on choisit de faire passer $T_h(\Gamma)$ par un contour fermé γ (à l'exception d'une arête) qui s'appuie sur le bord de Γ_0 et enveloppe toute cette surface à l'exception d'une bande permettant d'aller de Γ_e à Γ_s (cf. figure 4

Dans ces conditions, tous les degrés de liberté relatifs à des arêtes intérieures à γ sont a priori nuls puisque γ est le bord d'une surface Σ_γ où g est nulle. Nous posons $\tilde{\Gamma} = \Gamma \setminus \Sigma_\gamma$ et il suffit de construire $T_h(\Gamma)$ parmi l'ensemble des arêtes du maillage situées sur la portion $\tilde{\Gamma}$ de la frontière. De façon pratique, on retire toutes les arêtes de Γ_0 sauf une famille reliant Γ_e à Γ_s (arêtes en double trait sur la figure 4). Nous remarquons que pour chacune de ces arêtes, les degrés de liberté sont égaux (au signe près) au débit total traversant la tuyère (il suffit d'intégrer $\vec{\psi}$ le long du chemin OADE qui limite la face d'entrée par exemple). Ensuite on retire un arbre maximal dans l'ensemble des arêtes internes au maillage de la face OADE, et on enlève enfin une dernière arête liant cet arbre à Γ_0 . On procède de même pour la face de sortie Γ_s (arêtes en trait fort sur la figure 4).

Les arêtes engendrant $Y_h(\Omega)$ une fois repérées, le problème de Laplace-Beltrami est facile à résoudre. Il y a autant d'équations que d'éléments de surface k sur $\tilde{\Gamma}$:

$$(2.2) \quad \sum_{a \subset \partial k} \sigma_a(\bar{\psi}) = \int_k g \, d\gamma \quad \forall k \in \tilde{\Gamma}$$

et (2.2) se résout par substitution de proche en proche (cf. figure 4).

Pour construire l'espace $K_h(\Omega)$, on retire à l'ensemble des arêtes qui ne sont pas sur la frontière un arbre interne $T_h(\overset{\circ}{\Omega})$ qu'on relie à Γ_0 par une arête $T_h(\Gamma)$ (cf. Chapitre 2, Définitions 4.4 et 4.5). On enlève donc, pour chaque face située dans un plan du maillage $z = \text{cste}$, un arbre parmi les arêtes qui ne touchent pas Γ_0 , puis on joint ces plans entre eux par des arêtes parallèles à Oz , et l'ensemble $T_h(\overset{\circ}{\Omega})$ ainsi obtenu est relié à Γ_0 (cf. figure 5).

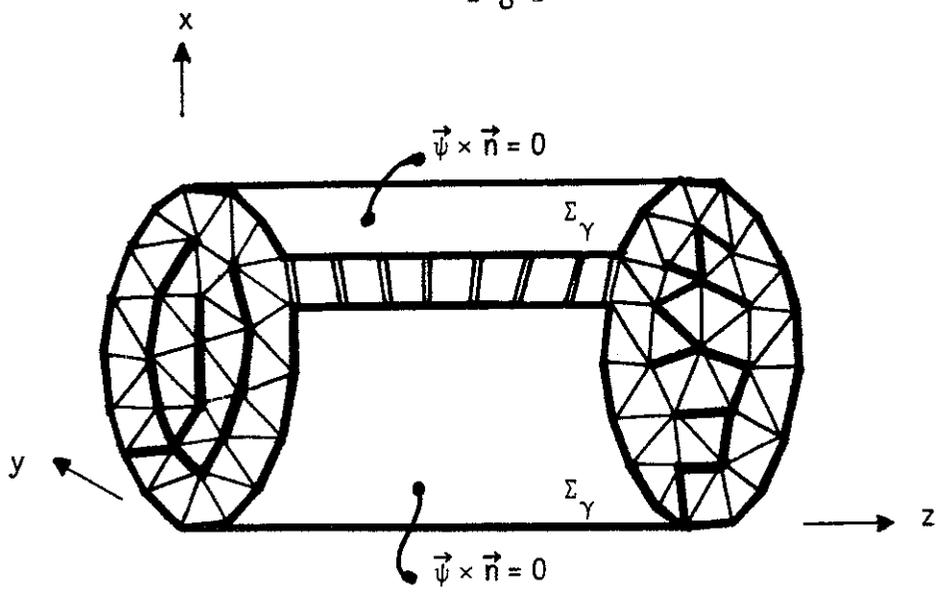


Figure 4 : Arbre surfacique sur $\tilde{\Gamma}$

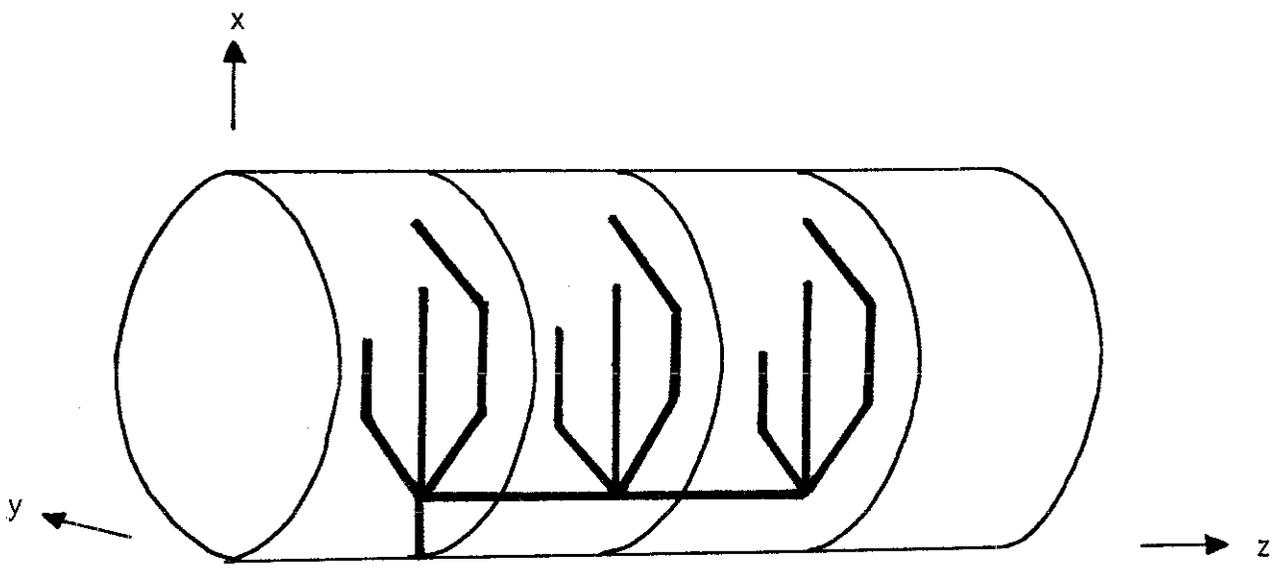


Figure 5 : Arbre interne

III - CALCUL DES MATRICES ELEMENTAIRES

1) PRISME DROIT

Nous étudions le système linéaire (1.7) posé dans l'espace $K_h(\Omega)$. Une base de cet espace est formée par les arêtes du maillage qui n'appartiennent ni à la frontière, ni à $T_h(\Omega)$. Nous rappelons d'abord le calcul des fonctions de base dans l'élément de référence \hat{K} puis nous proposons un calcul numérique de l'élément de matrice

$$(3.1) \quad a_{ij} = \int_K \text{rot } \vec{\varphi}_i \cdot \text{rot } \vec{\varphi}_j \, dx$$

où K désigne un prisme quelconque de \mathbf{R}^3 et $\vec{\varphi}_i$ la fonction de base associée à la i^{e} arête de K .

L'élément fini $(\hat{K}, \hat{\Sigma}, \hat{P})$ que nous avons utilisé est décrit dans NEDELEC [20] :

$$(3.2) \quad \begin{aligned} \hat{K} &= \{(\hat{x}, \hat{y}, \hat{z}), 0 \leq \hat{x}, \hat{y}, \hat{z} \leq 1, \hat{x} + \hat{y} \leq 1\} \\ \hat{\Sigma} &= \left\{ \sigma_a(\vec{\varphi}) \equiv \int_a \vec{\varphi} \cdot d\vec{l}, a \text{ arête de } \hat{K}, \text{ orientée selon la figure 6} \right\} \end{aligned}$$

$$(3.3) \quad \hat{P} = \left\{ \vec{\varphi} : \hat{K} \rightarrow \mathbf{R}^3, \exists \xi \in (P_1(z))^2, \exists \eta \in P_1(z), \exists \zeta \in P_1(x, y) / \right. \\ \left. \vec{\varphi}(\vec{x}) = \xi + \eta \vec{k} \times \vec{x} + \zeta \vec{k} \right\}$$

où \vec{k} désigne un vecteur unitaire selon Oz. On peut réécrire (3.3) sous la forme :

$$\vec{\varphi}(x, y, z) = \underbrace{\begin{pmatrix} a + cz \\ b + dz \\ 0 \end{pmatrix}}_{\vec{\xi}} + \underbrace{\begin{pmatrix} -(e + fz)y \\ (e + fz)x \\ 0 \end{pmatrix}}_{\eta \vec{k} \times \vec{x}} + \underbrace{\begin{pmatrix} 0 \\ 0 \\ \alpha + \beta x + \gamma y \end{pmatrix}}_{\zeta \vec{k}}$$

Nous déterminons d'abord la base $\hat{\varphi}_i$ de \hat{P} adaptée à l'élément fini $(\hat{K}, \hat{\Sigma}, \hat{P})$ étudié, i.e. telle que

$$(3.4) \quad \sigma_{a_i}(\hat{\varphi}_j) = \delta_{ij} \quad ; \quad i, j = 1, \dots, 9$$

puis nous calculons a_{ij} . Pour exprimer les vecteurs $\vec{r} \hat{\varphi}_i$, nous utilisons la base (\hat{p}_j) de l'espace \hat{P} liée naturellement à l'écriture

(3.3) :

$$\left\{ \begin{array}{llll} \hat{p}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} & \hat{p}_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} & \hat{p}_3 = \begin{pmatrix} z \\ 0 \\ 0 \end{pmatrix} & \hat{p}_4 = \begin{pmatrix} 0 \\ z \\ 0 \end{pmatrix} \\ & \hat{p}_5 = \begin{pmatrix} -y \\ x \\ 0 \end{pmatrix} & \hat{p}_6 = \begin{pmatrix} -yz \\ xz \\ 0 \end{pmatrix} & \\ \hat{p}_7 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} & \hat{p}_8 = \begin{pmatrix} 0 \\ 0 \\ x \end{pmatrix} & \hat{p}_9 = \begin{pmatrix} 0 \\ 0 \\ y \end{pmatrix} & \end{array} \right.$$

et M_{ij} la matrice de changement de base :

$$\hat{p}_j = \sum_{k=1}^9 M_{kj} \hat{\varphi}_k \quad , \quad \hat{\varphi}_i = \sum_{l=1}^9 (M^{-1})_{li} \hat{p}_l$$

donc (3.4) se réécrit : $M_{ij} = \int_{a_i} \hat{p}_j \cdot \vec{d}\vec{l}$ et il vient clairement :

$$M = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & -1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \end{pmatrix}$$

Par ailleurs, le calcul de $\text{rot } \hat{p}_j$ est élémentaire. On pose $(\text{rot } \hat{p}_j)_\alpha = R_{j\alpha}$ ($j = 1, 2, 3; \alpha = 1, 2, 3$). La matrice $R_{j\alpha}$ dépend du point $(x, y, z) \in \hat{K}$ et vaut :

$${}^tR = \begin{pmatrix} 0 & 0 & 0 & -1 & 0 & -x & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & -y & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 2 & 2z & 0 & 0 & 0 \end{pmatrix}$$

Enfin, le rotationnel des fonctions de base $\hat{\phi}_i$ est calculé par $(\text{rot } \hat{\phi}_i)_\alpha = P_{i\alpha}$ avec $P = {}^tM^{-1} \cdot R$:

$${}^tP = \begin{pmatrix} x & x & x-1 & -x & -x & 1-x & -1 & 0 & 1 \\ y-1 & y & y & 1-y & -y & -y & 1 & -1 & 0 \\ 2-2z & 2-2z & 2-2z & 2z & 2z & 2z & 0 & 0 & 0 \end{pmatrix}$$

L'évaluation numérique de a_{ij} grâce à la relation (3.1) se réduit alors à une simple intégration de polynômes sur \hat{K} .

2) PRISME QUELCONQUE

Dans le cas où l'élément K est un prisme de sommets A_1, \dots, A_6 , on le suppose image de \hat{K} par une application F biaffine (ZIENKIEWICZ [26] et figure 7) ;

$$(3.5) \quad \hat{K} \ni (\hat{x}, \hat{y}, \hat{z}) \mapsto F(\hat{x}) = \left[(1 - \hat{x} - \hat{y})A_1 + \hat{x} A_2 + \hat{y} A_3 \right] (1 - \hat{z}) + \left[(1 - \hat{x} - \hat{y})A_4 + \hat{x} A_5 + \hat{y} A_6 \right] \hat{z}$$

Les degrés de liberté Σ sur K sont définis par des intégrales sur chaque arête (relations (3.2)), et les fonctions d'interpolation sont de la forme

$$(3.6) \quad \vec{\phi}(x) = {}^t dF(\hat{x})^{-1} \cdot \hat{\phi}(\hat{x}), \hat{\phi} \in \hat{P}, x = F(\hat{x}) .$$

(cf. Chapitre 2, Définition 3.5).

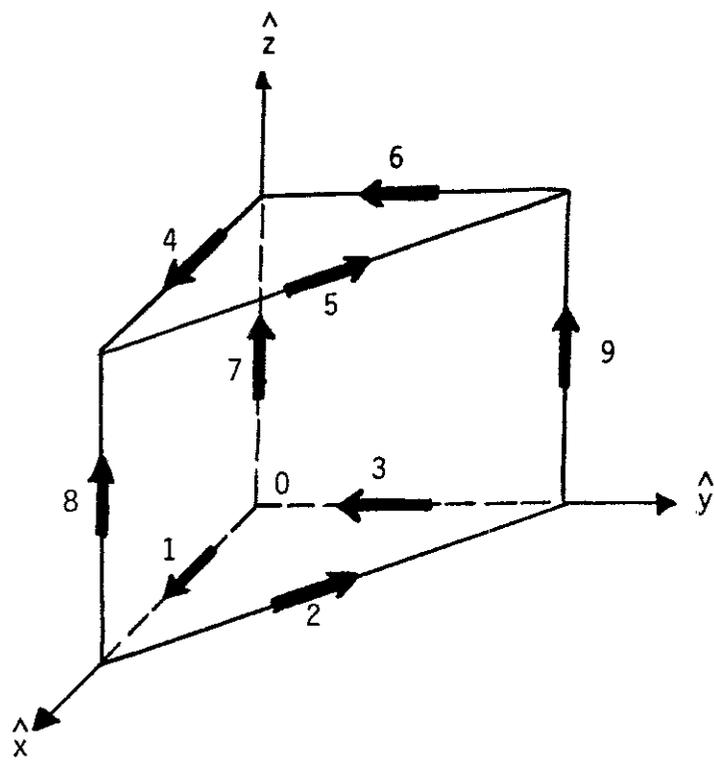


Figure 6 : prisme de référence \hat{K}

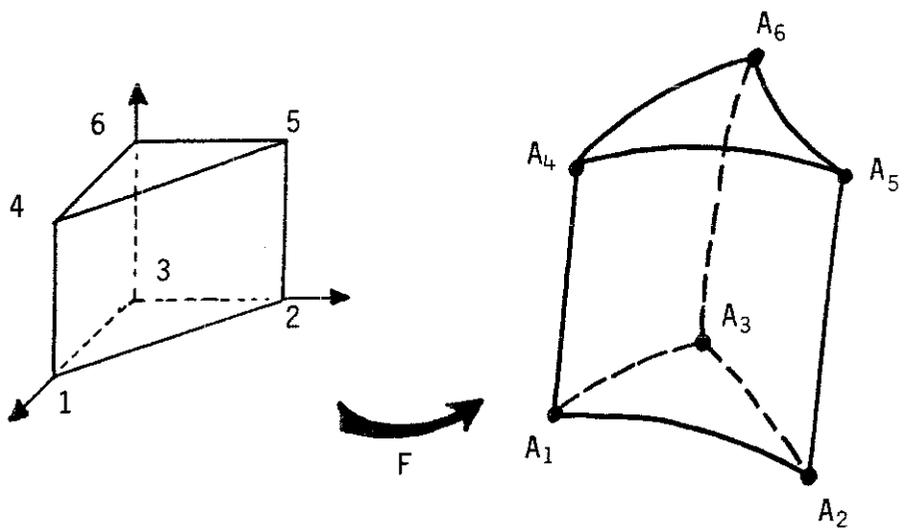


Figure 7 : Prisme curviligne

On note $\vec{\varphi}^i \equiv \vec{\varphi}_i$ l'image par (3.6) des fonctions de base $\hat{\varphi}_i$ définies par (3.4). Pour une fonction vectorielle $\vec{\varphi}$ quelconque, on pose :

$$(3.7) \quad \nabla \times \vec{\varphi}_{jk} = \sum_k \varepsilon_{ijk} (\text{rot } \vec{\varphi})_k$$

avec ε_{ijk} complètement antisymétrique (voir GERMAIN-MULLER [11] par exemple). On a alors aisément :

$$(3.8) \quad a_{ij} = \frac{1}{4} \sum_{k,l=1}^3 \int_K \left\{ (\nabla \times \vec{\varphi}^i)_{kl} (\nabla \times \vec{\varphi}^j)_{kl} - (\nabla \times \vec{\varphi}^i)_{kl} (\nabla \times \vec{\varphi}^j)_{lk} \right\} dx$$

Les deux termes de (3.8) sont égaux par antisymétrie de (3.7). Comme nous avons (Chapitre 2, Relation (3.33)) :

$$(\nabla \times \vec{\varphi})_{kl} = \sum_{m,n} dF_{mk}^{-1} (\nabla \times \hat{\varphi})_{mn} dF_{nl}^{-1} ,$$

nous en déduisons

$$\sum_{k,l} \nabla \times \vec{\varphi}_{kl}^i \cdot \nabla \times \vec{\varphi}_{lk}^j = \sum_{klmnpq} dF_{mk}^{-1} \cdot \nabla \times \hat{\varphi}_{mn}^i \cdot dF_{nl}^{-1} \cdot dF_{pl}^{-1} \cdot \nabla \times \hat{\varphi}_{pq}^j \cdot dF_{qk}^{-1}$$

soit, après un changement de variable (3.5) dans l'intégration sur K :

$$(3.9) \quad a_{ij} = -\frac{1}{2} \int_{\hat{K}} \text{tr} \left\{ D^2G \cdot (\nabla \times \hat{\varphi}_i) \cdot D^2G \cdot (\nabla \times \hat{\varphi}_j) \right\} J(\hat{x}) d\hat{x}$$

avec $D^2G = dF^{-1} \cdot {}^t dF^{-1}$ et $J(\hat{x}) = \det(dF(\hat{x}))$. Il suffit de poser :

$$Q^1 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{pmatrix}, \quad Q^2 = \begin{pmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \quad Q^3 = \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

pour réécrire (3.9) sous la forme :

$$a_{ij} = -\frac{1}{2} \int_{\hat{K}} \sum_{\alpha, \beta} P_{i\alpha} \text{tr} \left\{ D^2G \cdot Q^\alpha \cdot D^2G \cdot Q^\beta \right\} P_{j\beta} J(\hat{x}) d\hat{x} .$$

Cette dernière intégrale a été évaluée à l'aide d'une formule de quadrature comprenant 30 points dans l'élément de référence \hat{K} : 5 points sur $[0,1]$ (formules de Gauss, de degré 9, voir STROUD [25] par exemple), par 6 points dans le triangle unité ; nous rappelons les formules de Hammer de degré 4 utilisées dans ce dernier cas (cf. DHATT-TOUZO [4] par exemple et la figure 8) :

$$\iint_{\{0 \leq \hat{x} \leq 1, 0 \leq \hat{y} \leq 1, \hat{x} + \hat{y} \leq 1\}} f(\hat{x}, \hat{y}) d\hat{x} d\hat{y} \approx \sum_{j=1}^6 w_j f(\hat{x}_j, \hat{y}_j)$$

avec $w_j, \hat{x}_j, \hat{y}_j$ donnés par :

	\hat{x}_j	\hat{y}_j	\hat{w}_j
$a = 0,445948490915965$	a 1-2a a	a a 1-2a	} 0,111690794839005
$b = 0,091576213509771$	b 1-2b b	b b 1-2b	} 0,054975871827661

IV - RESOLUTION DU SYSTEME LINEAIRE

1) STOCKAGE DE LA MATRICE

La matrice A du système linéaire (1.7) est symétrique et définie positive puisque par construction l'application rotationnel est injective dans l'espace $K_h(\Omega)$. Pour le stockage de la matrice, d'ordre $N \times N$, on repère, pour chaque fonction de base (globale) de $K_h(\Omega)$ associée à l'arête numéro i , l'ensemble des arêtes de numéro $j > i$ voisines de l'arête i , c'est-à-dire telles que $a_{ij} \neq 0$ a priori. On construit de cette façon un pointeur $NVOIS(.,.)$ à double entrée, qui associe au couple (i,k) le numéro d'arête de la $k^{\text{ème}}$ voisine de l'arête i . Ce tableau est de dimension $N \times 17$ puisqu'avec la numérotation utilisée, une arête donnée a au plus 17 voisins (cf. figure 9).

L'assemblage de la matrice A doit être effectué avec attention, puisque c'est l'orientation globale des arêtes qui importe, alors que le calcul élémentaire du paragraphe 3 fait intervenir les orientations particulières de l'élément de référence.

On stocke ensuite l'élément a_{ij} pour j $k^{\text{ème}}$ voisin de l'arête i dans un tableau réel $N \times 18$ (compte tenu des éléments diagonaux). De cette façon, un produit du type $y = A.x$ peut s'écrire :

$$\begin{array}{l} \text{Initialisation ; boucle sur } i : 1 \rightarrow N \\ \left[\begin{array}{l} y(i) = A(i,18) * x(i) \\ \text{boucle sur les voisins ; } k : 1 \rightarrow 17 \\ \left[\begin{array}{l} \text{boucle sur les arêtes } i : 1 \rightarrow N \\ \left[\begin{array}{l} (4.1) \quad y(i) \leftarrow y(i) + A(i,k) * x(NVOIS(i,k)) \\ \text{boucle sur les arêtes } i : 1 \rightarrow N \\ \left[\begin{array}{l} (4.2) \quad y(NVOIS(i,k)) \leftarrow y(NVOIS(i,k)) + A(i,k) * x(i) \end{array} \right. \end{array} \right. \end{array} \right. \end{array} \right. \end{array} \end{array}$$

La boucle (4.1) se calcule de façon vectorielle grâce au sous programme GATHER (cf. CCVR [2]) et la boucle (4.2) peut se vectoriser grâce au SCATTER lorsque l'application $i \mapsto \text{NVOIS}(i,k)$ est injective (DUPUY [7]). Nous avons vectorisé la première boucle (4.1), et gagné ainsi un facteur 3 en temps calcul sur l'ensemble du programme.

2) ALGORITHME DU GRADIENT CONJUGUE

La résolution du système linéaire $Ax=b$ est effectuée grâce à la méthode du gradient conjugué préconditionné (HESTENES-STIEFEL [13], EVANS [9], AXELSSON [1]) décrit aussi dans GOLUB-MEURANT [12] et LASCAUX-THEODOR [16]. Nous la rappelons brièvement : on se donne une matrice S de préconditionnement telle que la résolution du système linéaire $S.x=b$ soit faible.

ALGORITHME

(i) *Initialisation*

$$x^0 \in \mathbb{R}^N, g^0 = Ax^0 - b, w^0 = S^{-1} g^0, v^0 = w^0, z^0 = Ax^0.$$

(ii) *Itération* : x^n, g^n, w^n, v^n, z^n sont connus

$$(4.3) \quad z^{n+1} = A.w^n$$

$$\rho_{n+1} = - \frac{(g^n, w^n)}{(z^{n+1}, w^n)}$$

$$x^{n+1} = x^n + \rho_{n+1} w^n$$

$$g^{n+1} = A.x^{n+1} - b = g^n + \rho_{n+1} z^{n+1}$$

$$(4.4) \quad v^{n+1} = S^{-1} g^{n+1}$$

$$\gamma_{n+1} = \frac{(v^{n+1}, g^{n+1})}{(w^n, g^n)}$$

$$w^{n+1} = S^{-1} g^{n+1} + \gamma_{n+1} w^n = v^{n+1} + \gamma_{n+1} w^n$$

(iii) Test de fin : on compare $\|g^{n+1}\|_2 / \|g\|_2$ à $\varepsilon > 0$ donné d'avance.

Nous observons qu'avec cette écriture de l'algorithme, un seul produit du type matrice par vecteur est nécessaire (équation (4.3)) ainsi qu'une seule résolution du système de préconditionnement (4.4). Nous avons successivement envisagé trois préconditionneurs classiques, classés par difficulté croissante de mise en oeuvre :

$$(4.5) \quad S = \text{diag}(A)$$

$$(4.6) \quad S = \text{préconditionneur SSOR}$$

$$(4.7) \quad S = \text{décomposée incomplète de Cholesky}$$

qui ont été testés dans un cas linéaire, pour une simple boîte cubique au lieu du GAMM channel, avec un test d'arrêt à $\varepsilon = 10^{-4}$. Tout d'abord, il est indispensable de faire les calculs avec un processeur 64 bits.

Le préconditionneur diagonal engendre en général de nombreuses itérations du gradient conjugué, mais la résolution de (4.4) pour ce choix (4.5) est très économique car naturellement vectorielle. Son emploi s'est avéré assez satisfaisant. Le choix (4.6), plus précisément

$$S = \frac{1}{2-\omega} (L + \frac{\Delta}{\omega}) (\frac{\Delta}{\omega})^{-1} (\frac{\Delta}{\omega} + {}^tL)$$

avec $A = L + \Delta + {}^tL$, L triangulaire inférieure à diagonale nulle et $\omega = 1.5$, est une variante de la méthode de STONE [24] et DUPONT-KENDALL-RACHFORD [6]

qui conduit à un nombre d'itérations plus faible (trois fois moins typiquement) mais moins économique qu'avec le choix diagonal, à cause des descentes-remontées pour résoudre (4.4). Les tests effectués nous ont montré qu'en temps machine, elle est plus économique d'environ 5 % par rapport au choix (4.5).

La décomposition incomplète de Cholesky (4.7) (MEIJERINK-VAN DER VORST [17], KERSHAW [15]) a été décevante, puisqu'au cours de la décomposition, nous avons parfois obtenu des pivots négatifs. Ceci n'est en fait pas étonnant, car rien n'assure que A soit une M -matrice, condition suffisante de construction de l'algorithme ([17]). Par ailleurs, même lorsqu'aucun pivot négatif n'apparaît, le temps CPU reste comparable à celui obtenu pour le préconditionneur SSOR, qui a finalement été retenu pour les calculs non-linéaires.

3) INFLUENCE DE L'ARBRE INTERNE

La dépendance a priori du système linéaire à résoudre avec le choix de l'arbre interne $T_h(\Omega)$ nous a conduit à étudier son influence pour le même cas test que précédemment (problème linéaire, boîte cubique), avec un préconditionnement SSOR. Le faible nombre de degrés de liberté pour ce test ($N = 249$) nous a permis de calculer facilement toutes les valeurs propres de la matrice A . Nous avons porté figure 10 la forme de l'arbre utilisé (cf. aussi figure 5), le conditionnement $\lambda_{\max}/\lambda_{\min}$ de A , ainsi que le nombre d'itérations du gradient conjugué nécessaires pour atteindre un résidu $\epsilon = 10^{-4}$. Aucune conclusion simple ne se dégage : un choix d'arbre par arêtes verticales (tests nos 2 à 7) donne les meilleurs comme les

pires conditionnements, et le simple changement d'une seule arête (tests 3,4,5) peut changer de façon sensible le conditionnement comme le nombre d'itérations ; un choix d'arbre qui privilégie les arêtes horizontales (tests 1,8,9) a donné les meilleures performances, malgré la forte variation du conditionnement. Comme nous n'avons pas étudié de façon plus précise la variation du spectre de A avec l'arbre $T_h(\Omega)$ (JENNINGS [14]), nous avons conservé notre choix initial (test 1 en figure 5), en renforçant la précision lors des calculs non linéaires ($\epsilon = 10^{-6}$ à 10^{-8}).

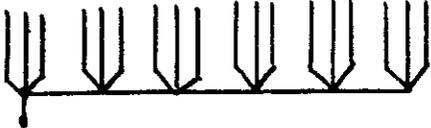
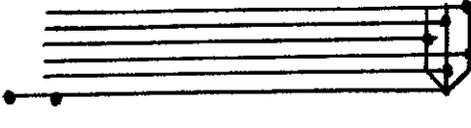
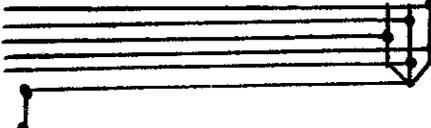
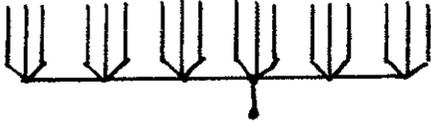
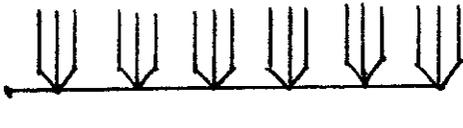
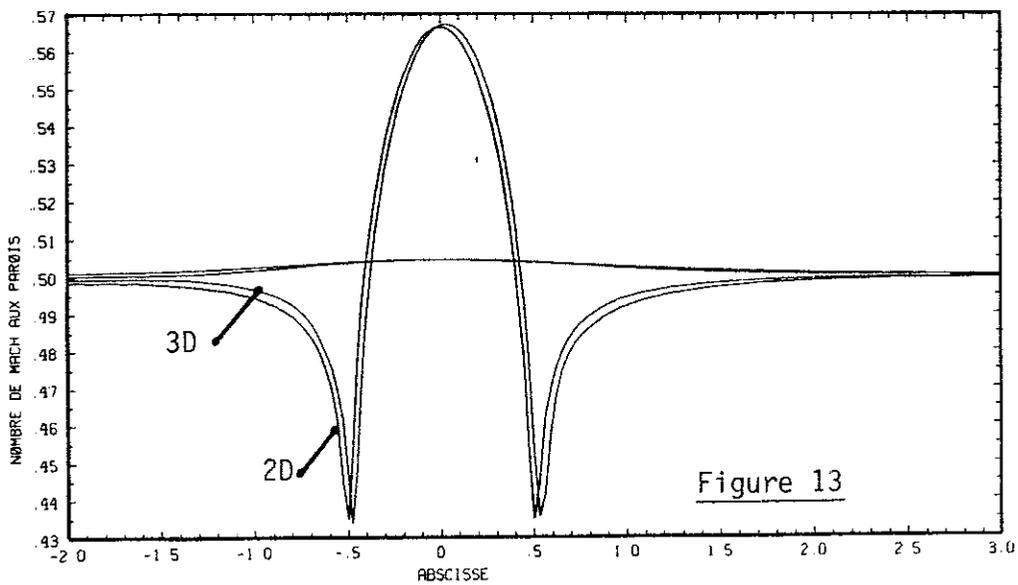
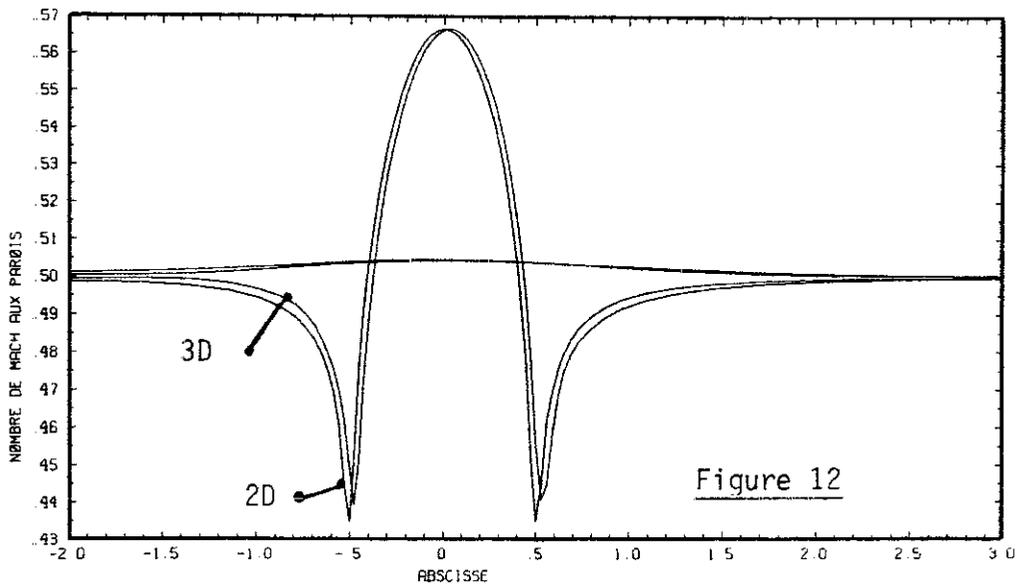
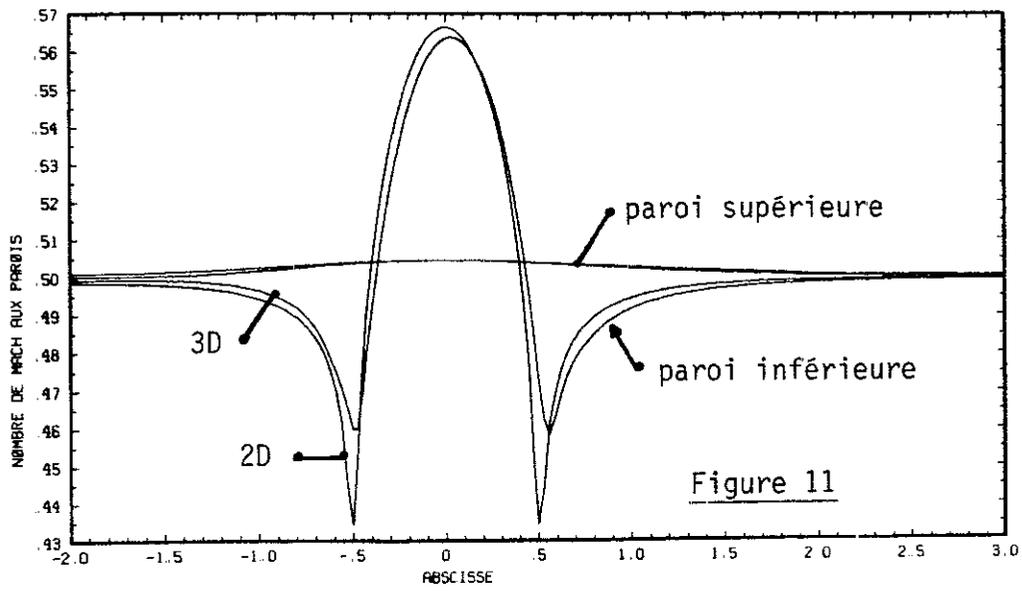
	$\frac{\lambda_{\max}}{\lambda_{\min}}$	nombre d'itération: ($\epsilon = 10^{-4}$)
 1	14150	40
 2	6811	74
 3	16096	70
 4	4062	59
 5	22169	71
 6	34045	84
 7	28763	86
 8	6545	44
 9	18360	42

Figure 10 : Test des arbres internes

V - TESTS NUMERIQUES, CONCLUSION

Nous avons testé la méthodologie décrite dans les paragraphes précédents pour le GAMM channel, à un nombre de Mach amont $M_0 = 0,50$. Le maillage de la face transversale OADE (figure 3) comporte trois couches de points, soit 80 triangles au total. L'ensemble des arêtes internes utiles pour le système linéaire s'élève à $N = 6879$. La convergence de l'algorithme de point fixe pour la résolution de la non linéarité est toujours rapide (moins de 10 itérations typiquement) et demande 5 mn CPU environ sur le Cray 1S du CCVR. Nous avons comparé ce cas test faussement tridimensionnel à un code bidimensionnel existant (cf. Chapitre 4). Dans un premier cas de calcul, nous avons choisi l'épaisseur $Y = OE$ égale à 1. Les résultats relatifs à la face OABC sont portés figure 11, où l'on a tracé les valeurs du nombre de Mach sur les lignes OC et AB respectivement. Puis nous avons resserré le canal en choisissant $Y = 0,05$ afin de diminuer les effets tridimensionnels parasites du cas précédent. Une amélioration importante dans la qualité des résultats est à noter (figure 12). Enfin, toutes choses égales par ailleurs, nous avons changé la direction des arêtes diagonales dans le maillage de la face d'entrée (figure 3) ; les résultats (figure 13) sont comparables avec ceux du cas précédent, mais le biais introduit par le choix du maillage triangulaire est encore visible.

Nous avons présenté un test numérique de l'élément conforme dans $H(\text{rot})$ à base prismique de J.C. Nédélec [20] pour un problème faiblement non-linéaire de fluide parfait compressible dans un canal. La mise en oeuvre



Nombre de Mach sur les parois du canal. Comparaison des calculs bidimensionnel (2D) et tridimensionnels (3D) pour trois maillages différents de la face d'entrée.

de la jauge axiale discrète n'a pas engendré de conditionnement trop grand pour le système linéaire à résoudre, et la méthode du gradient préconditionné par SSOR s'est avérée performante. La faible mémoire du Cray 1S nous a conduit à utiliser un maillage tridimensionnel semi-déductif, donc à écrire un code très spécialisé. Par contre, le logiciel très général développé parallèlement à notre travail par J.M. Dupuy devrait permettre de nouveaux tests de validation (sur Cray 2) de la méthode dans des géométries complètement tridimensionnelles.

. Juin 1987 .

REFERENCES

- [1] AXELSSON O., *Solution of Linear Systems of Equations : Iterative Methods*, Lect Notes in Math n° 572, Copenhagen 1976. Springer Verlag, Berlin 1977.
- [2] CCVR : Cours de Vectorisation. Centre de Calcul Vectoriel pour la Recherche, Palaiseau, 1985.
- [3] CIAVALDINI J.F. Thèse d'Etat, Rennes, 1980.
- [4] DHATT G., TOUZO G., Une présentation de la méthode des éléments finis, Maloine, Paris, 1984.
- [5] DUBOIS F., DUPUY J.M., *A Three Dimensional Vector Potential Formulation for Solving Transonic Flows with Mixed Finite Elements*, 6° International Symposium : Finite Element Methods in Flow Problems, Inria, Antibes, 1986.
- [6] DUPONT T., KENDALL R., RACHFORD H., *An Approximate Factorization Procedure for Solving Self-Adjoint Elliptic Difference Equations*, SIAM J. Numer Anal, vol 5, n° 3, pp 559-573, 1968.
- [7] DUPUY J.M., *Ecoulements Transsoniques Potentiels Tridimensionnels*, Contrat DRET n° 81/228, Annexe 2, avril 1985.
- [8] DUPUY J.M., *Résolution Numérique des Equations d'Ecoulements Compressibles Transsoniques Tridimensionnels Stationnaires*, Contrat DRET n° 83/434, Rapport de synthèse finale, juillet 1986.
- [9] EVANS D.J., *The Use of Pre-conditioning in Iterative Methods for Solving Linear Equations with Symmetric Definite Matrices*, J Inst Maths Applics 4, pp 295-314, 1967.
- [10] GELDER D., *Solution of the Compressible Flow Equations*, Int J. for Num Meths in Eng, vol 3, pp 35-43, 1971.
- [11] GERMAIN P., MULLER P., Introduction à la Mécanique des milieux continus, Masson, Paris, 1980.
- [12] GOLUB G.H., MEURANT G., Résolution Numérique des grands Systèmes Linéaires, Eyrolles, Paris, 1984.
- [13] HESTENES M., STIEFEL E., *Methods of Conjugate Gradients for Solving Linear Systems*, J. of Research Nat Bur of Standards, vol 49, n° 6, pp 409-436, 1952.

- [14] JENNINGS A., *Influence of the Eigenvalue Spectrum on the Convergence Rate of the Conjugate Gradient Method*, J Inst Maths Applics, vol 20, pp 61-72, 1977.
- [15] KERSHAW D.S., *The Incomplete Cholesky-Conjugate Gradient Method for the Iterative Solution of Systems of Linear Equations*, J of Comp Phys, vol 26, pp 43-65, 1978.
- [16] LASCAUX P., THEODOR R., *Analyse Numérique Matricielle Appliquée à l'Art de l'Ingénieur*, Masson, Paris, 1986.
- [17] MEIJERINK J.A., VAN DER VORST H.A., *An Iterative Solution Method for Linear Systems of Which the Coefficient Matrix is a Symmetric M-Matrix*, Math of Comp, vol 31, n° 137, pp 148-162, 1977.
- [18] NEČAS J., *Compacité par entropie et écoulements de fluides*, Masson, Paris, à paraître.
- [19] NEDELEC J.C., *Mixed Finite Elements in \mathbb{R}^3* , Numer Math 35, pp 315-341, 1980.
- [20] NEDELEC J.C., *Elements Finis*, Cours polycopié, Ecole Nationale des Ponts et Chaussées, Paris, 1982.
- [21] PERRONNET A., *Mefisto maillage, Manuel d'utilisation*, Laboratoire d'Analyse Numérique, Université Paris 6, 1983.
- [22] RIZZI A., VIVIAND H., (Eds) *Numerical Methods for the Computation of Inviscid Transonic Flows with Shock Waves*, Notes on Numer Fluid Mechanics, vol 3, Vieweg, Braunschweig, 1981.
- [23] ROUX F.X., Séminaire, Laboratoire d'Analyse Numérique de l'Université Paris 6, mars 1984.
- [24] STONE H.L., *Iterative Solution of Implicit Approximations of Multi-dimensional Partial Differential Equations*, SIAM J. Numer Anal, vol 5, n° 3, pp 530-558, 1968.
- [25] STROUD A.H., *Approximate Calculation of Multiple Integrals*, Prentice-Hall, Inc, Englewood Cliffs, 1971.
- [26] ZIENKIEWICZ O.C., *The Finite Element Method in Engineering Science*, Mc Graw-Hill, London, 1971.