

DANIEL TURGEON

**ADAPTATION DE MAILLAGES
ET
ÉTUDE D'UN ESTIMATEUR D'ERREUR *A POSTERIORI*
POUR LES FLUIDES VISCOÉLASTIQUES**

Mémoire présenté
à la Faculté des Études Supérieures
pour l'obtention
du grade de Maître ès Sciences (M.Sc.)

Département de Mathématiques et Statistique
FACULTÉ DES SCIENCES ET GÉNIE
UNIVERSITÉ LAVAL

MAI 2001

©Daniel Turgeon, 2001



**National Library
of Canada**

**Acquisitions and
Bibliographic Services**

**395 Wellington Street
Ottawa ON K1A 0N4
Canada**

**Bibliothèque nationale
du Canada**

**Acquisitions et
services bibliographiques**

**395, rue Wellington
Ottawa ON K1A 0N4
Canada**

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-61364-X

Canada

Résumé

Ce mémoire est consacré à l'étude de techniques de simulation numérique qui combinent la méthode de discrétisation par éléments finis et les idées d'adaptation. Il existe de nombreux champs d'application de telles techniques, mais nous nous sommes concentrés sur la simulation des fluides dits non-newtoniens qui sont d'un intérêt particulier pour l'industrie des plastiques.

Le modèle viscoélastique de Maxwell est souvent considéré comme un bon banc d'essai et c'est celui que nous avons retenu. Pour l'outil informatique, nous avons choisi de travailler avec le code MEF++ développé à l'université Laval. Ce code est suffisamment souple et son utilisation nous a permis de monter assez rapidement une librairie d'outils d'adaptation performants. Nos différents tests démontrent clairement le potentiel de cette approche pour l'étude numérique de ces fluides complexes.

Daniel Turgeon

Directeur : Roger Pierre

Avant-propos

Tout d'abord, j'aimerais souligner la grande disponibilité et la compréhension de mon directeur M. Roger Pierre ainsi que de mon codirecteur M. Robert Guénette, dont ils ont fait preuve pendant les deux longues années durant lesquelles j'ai rédigé ce mémoire à temps partiel et à distance.

Je voudrais également remercier mon épouse Julie Labbé, pour m'avoir donné le courage et la force de mener à terme cet ouvrage. Je dois aussi remercier ma fille Laurianne, pour toute la sagesse qu'elle a su démontrer pour me permettre de rédiger ce mémoire. Ensuite, je tiens à remercier mes parents Micheline et Claude ainsi que mon frère Benoit qui ont toujours su m'encourager et me supporter dans tous mes projets. Je remercie aussi Françoise et Jean-Guy pour s'être occupés de Laurianne durant mes multiples visites à l'université.

Enfin, je remercie tous ceux qui ont cheminé avec moi durant mes études et qui ont rendu ces années si agréables : Julie Labbé à qui je dédie ce mémoire, Patrick Pilot, Martin Busque, Charles Fortin, Julie Pelchat, Manon Drolet et Hans Bhérer. Un gros merci à Carl Robitaille et Éric Chamberland pour tout le support qu'ils m'ont donné sur le logiciel MEF++.

Table des matières

Avant-propos	i
Table des matières	ii
Table des figures	vi
Liste des tableaux	xi
Introduction	1
1 Les méthodes adaptatives en éléments finis	5
1.1 Le problème modèle de Poisson	5
1.2 Discrétisation du problème de Poisson	7
1.3 Adaptation	11
2 Estimation d'erreur	12
2.1 Estimation d'erreur <i>a priori</i>	13
2.2 Estimation d'erreur <i>a posteriori</i>	15
2.2.1 Estimateurs d'erreur <i>a posteriori</i>	15

2.2.2	Approximation des dérivées du deuxième ordre d'une fonction linéaire par morceaux	16
2.2.3	Fonction de raffinement	18
2.2.4	Raffinement anisotrope	19
2.2.5	Calcul de la longueur d'une arête dans la métrique	20
2.2.6	Erreur cible et tolérance	20
2.2.7	Qualité des éléments	21
2.2.8	Quelques remarques	22
3	L'adaptation de maillages	23
3.1	Retournement d'arêtes	25
3.1.1	Retournement d'une arête interne	25
3.1.2	Retournement d'une arête de bord	27
3.2	Déplacement de sommets	28
3.3	Raffinement d'arêtes	30
3.3.1	Raffinement d'une arête interne	32
3.3.2	Raffinement d'une arête de bord	32
3.4	Déraffinement d'arêtes	34
3.4.1	Algorithme de reconstruction d'une coquille vide	37
3.5	Séquence d'adaptation	40
4	Implémentation d'un code d'adaptation en C++	42
4.1	Le GIREF et MEF++	42
4.2	Réutilisation des bibliothèques de MEF++	44

4.3	Développement en C++	45
4.3.1	Modification et création de bibliothèques	45
4.3.2	Modification et création de programmes	47
4.3.3	Fichier de paramètres pour l'adaptation	48
4.4	Validation du code	50
4.4.1	Géométrie et maillage de départ	51
4.4.2	Adaptation de référence et fichier de paramètres	53
4.4.3	Validation du déraffinement et des boucles de nettoyage	54
4.4.4	Validation de l'amélioration de la qualité des éléments	58
4.4.5	Validation du critère de la longueur euclidienne minimale	60
4.4.6	Validation de la séquence d'adaptation	61
4.4.7	Conclusion sur les tests	66
5	Applications aux fluides viscoélastiques	67
5.1	Les principes fondamentaux de la mécanique des fluides	67
5.1.1	La loi de conservation de la masse et les fluides incompressibles	67
5.1.2	La loi de conservation de la quantité de mouvement	69
5.2	Le système de Stokes pour les fluides incompressibles	72
5.3	Équations d'évolution des fluides viscoélastiques	73
5.4	Discrétisation des équations	76
5.4.1	Résolution numérique du problème de Stokes	76
5.4.2	Problème global	80
5.4.3	Les méthodes de décentrage selon les lignes de courant	81

5.4.4	Discretisation de la loi de comportement de Maxwell	83
5.5	Résultats numériques	84
5.5.1	Écoulement de Poiseuille	85
5.5.2	Résolution du problème de Poiseuille : $\lambda = 0.0$	86
5.5.3	Résolution du problème de Poiseuille : $\lambda = 1.0$	90
5.5.4	Problème de la couche limite	94
5.5.5	Résolution du problème de la couche limite : $\lambda = 0.0$	95
5.5.6	Résolution du problème de la couche limite : $\lambda = 1.0$	100
5.5.7	Problème de l'écoulement autour d'un cylindre	109
5.5.8	Résolution de l'écoulement autour d'un cylindre : $\lambda = 0.0$	110
5.5.9	Résolution de l'écoulement autour d'un cylindre : $\lambda = 1.0$	116
5.5.10	Problème de la contraction	122
5.5.11	Résolution du problème de la contraction : $\lambda = 0.0$	124
5.5.12	Résolution du problème de la contraction : $\lambda = 1.0$	131
	Conclusion	139
	Bibliographie	141

Table des figures

1.1	Coquille extraite d'un maillage.	10
3.1	Retournement d'une arête interne.	27
3.2	Retournement d'une arête de bord.	28
3.3	Déplacement d'un sommet.	30
3.4	Raffinement d'une arête interne.	33
3.5	Raffinement d'une arête de bord.	34
3.6	Déraffinement d'un sommet.	37
4.1	Géométrie utilisée pour la validation.	51
4.2	Maillage de base (nommé ORIGINAL) utilisé pour la validation.	52
4.3	Fonction analytique de l'erreur (4.1).	52
4.4	Norme de la métrique.	53
4.5	Maillage RÉFÉRENCE obtenu avec l'adaptation dite de référence.	55
4.6	Agrandissement du maillage RÉFÉRENCE.	55
4.7	Norme de σ calculée sur le maillage de référence.	56
4.8	Norme de σ calculée sur un maillage contenant des éléments très plats. . .	56
4.9	Maillage DÉRAF1 obtenu pour la validation du déraffinement.	59

4.10	Agrandissement du maillage DÉRAF1, coin supérieur droit.	59
4.11	Agrandissement du maillage DÉRAF2, coin supérieur droit.	60
4.12	Agrandissement du maillage RAFDÉRAF obtenu en enlevant les séquences de nettoyage.	61
4.13	Agrandissement du maillage RAFFINEMENT obtenu en posant l_{min}^e assez grand.	62
4.14	Agrandissement du maillage TestCritereLongEuclMinAvantRet.	62
4.15	Agrandissement du maillage TestCritereLongEuclMinApresRet.	63
4.16	Agrandissement du maillage INVERSION obtenu en inversant raffinement et déraffinement.	66
5.1	Représentation schématique du système physique de Maxwell.	74
5.2	Géométrie du canal.	85
5.3	Maillage ORIGINAL du canal.	87
5.4	Maillages ADAPT1, ADAPT2 et ADAPT3 (de haut en bas) générés pour le problème de Poiseuille : Cas $\lambda = 0.0$	89
5.5	L'erreur $\ (\tau_h)_{11} - (\pi_h \tau)_{11}\ _\infty$ sur le maillage SANS DER.	91
5.6	Maillages ADAPT1, ADAPT2, ADAPT3 et SANS DER (de haut en bas) générés pour le problème de Poiseuille : Cas $\lambda = 1.0$	92
5.7	Maillages ADAPT1, ADAPT2, ADAPT3 et REG (de haut en bas) générés pour le problème de couche limite : Cas $\lambda = 0.0$	99
5.8	Maillages ADAPT1, ADAPT2, ADAPT3 et ADAPT4 (de haut en bas) générés pour le problème de couche limite : Cas $\lambda = 1.0$	105
5.9	Agrandissement du maillage ADAPT4 (coin supérieur gauche) généré pour le problème de couche limite : Cas $\lambda = 1.0$	106

5.10 Coupe de la solution τ_{11} et τ_{12} en $x=2.5$: Cas $\lambda = 1.0$ du problème de couche limite. 106

5.11 Solution τ_{11} : Cas $\lambda = 1.0$ du problème de couche limite. 107

5.12 Solution τ_{12} : Cas $\lambda = 1.0$ du problème de couche limite. 107

5.13 Courbes de niveau de τ_{11} , coin supérieur gauche : Cas $\lambda = 1.0$ du problème de couche limite. 108

5.14 Courbes de niveau de u_x : Cas $\lambda = 1.0$ du problème de couche limite. . . . 108

5.15 Courbes de niveau de p : Cas $\lambda = 1.0$ du problème de couche limite. . . . 109

5.16 Géométrie du canal avec obstacle. 109

5.17 Maillage ORIGINAL utilisé pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 0.0$ 111

5.18 Maillages ADAPT1, ADAPT2 et ADAPT3 (de haut en bas) générés pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 0.0$ 113

5.19 Agrandissement du maillage ADAPT3 (aux alentours de l'obstacle) généré pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 0.0$ 113

5.20 De haut en bas, les courbes de niveau des variables τ_{11} , τ_{12} et τ_{22} et par la suite les graphes des variables τ_{11} , τ_{12} et τ_{22} sur ADAPT3 pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 0.0$ 114

5.21 Courbes de niveau de p sur ADAPT3 pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 0.0$ 115

5.22 Graphe de la variable u sur ADAPT3 autour de l'obstacle pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 0.0$ 115

5.23 Maillages ADAPT1, ADAPT2, ADAPT3 et ADAPT4 (de haut en bas) générés pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 1.0$. 119

5.24 Agrandissement du maillage ADAPT4 (aux alentours de l'obstacle) généré pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 1.0$ 119

5.25 Courbes de niveau des variables τ_{11} , τ_{12} et τ_{22} (de haut en bas) sur ADAPT4 pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 1.0$ 120

5.26 Graphe des variables τ_{11} , τ_{12} , τ_{12} et τ_{22} (de haut en bas) sur ADAPT4 pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 1.0$ 121

5.27 Graphe et courbes de niveau de la variable p sur ADAPT4 pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 1.0$ 121

5.28 Graphe de la variable u sur ADAPT4 autour de l'obstacle pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 1.0$ 122

5.29 Géométrie de la contraction. 123

5.30 Maillage ORIGINAL utilisé pour le problème de la contraction : Cas $\lambda = 0.0$ 124

5.31 Maillage ADAPT1 généré pour le problème de la contraction : Cas $\lambda = 0.0$. 126

5.32 Maillage ADAPT2, vu sous différents agrandissements, pour le problème de la contraction : Cas $\lambda = 0.0$ 126

5.33 De haut en bas, les courbes de niveau des variables τ_{11} , τ_{12} et τ_{22} et par la suite les graphes des variables τ_{11} , τ_{12} et τ_{22} sur ADAPT2 pour le problème de la contraction : Cas $\lambda = 0.0$ 127

5.34 Graphe de la variables p selon 2 vues différentes sur ADAPT2 pour le problème de la contraction : Cas $\lambda = 0.0$ 128

5.35 Graphe de la variable u sur ADAPT2 autour de l'obstacle pour le problème de la contraction : Cas $\lambda = 0.0$ 128

5.36 Coupe en $y = 1$ passant par la singularité pour τ_{11} sur ADAPT2 pour le problème de la contraction : Cas $\lambda = 0.0$ 129

5.37 Coupe en $y = 1$ passant par la singularité pour τ_{12} sur ADAPT2 pour le problème de la contraction : Cas $\lambda = 0.0$ 129

5.38 Coupe en $y = 1$ passant par la singularité pour τ_{22} sur ADAPT2 pour le problème de la contraction : Cas $\lambda = 0.0$ 130

5.39 Coupe en $y = 0$ pour τ_{12} sur ADAPT2 pour le problème de la contraction : Cas $\lambda = 0.0$ 130

5.40 Agrandissements des maillages ADAPT1, ADAPT2, ADAPT3 et ADAPT4 (de haut en bas) générés pour le problème de la contraction : Cas $\lambda = 1.0$. 133

5.41 Agrandissement du maillage ADAPT4 près du coin (0,1) pour le problème de la contraction : Cas $\lambda = 1.0$ 133

5.42 De haut en bas, les courbes de niveau des variables τ_{11} , τ_{12} et τ_{22} et par la suite les graphes des variables τ_{11} , τ_{12} et τ_{22} sur ADAPT4 pour le problème de la contraction : Cas $\lambda = 1.0$ 134

5.43 Graphe de la variables p selon 2 vues différentes sur ADAPT4 pour le problème de la contraction : Cas $\lambda = 1.0$ 135

5.44 Graphe de la variable u sur ADAPT4 autour de l'obstacle pour le problème de la contraction : Cas $\lambda = 1.0$ 135

5.45 Coupe en $y = 1$ passant par la singularité pour τ_{11} sur ADAPT4 pour le problème de la contraction : Cas $\lambda = 1.0$ 136

5.46 Coupe en $y = 1$ passant par la singularité pour τ_{12} sur ADAPT4 pour le problème de la contraction : Cas $\lambda = 1.0$ 136

5.47 Coupe en $y = 1$ passant par la singularité pour τ_{22} sur ADAPT4 pour le problème de la contraction : Cas $\lambda = 1.0$ 137

5.48 Coupe en $y = 0$ pour τ_{12} , τ_{12} et τ_{22} (De haut en bas) sur ADAPT4 pour le problème de la contraction : Cas $\lambda = 1.0$ 138

Liste des tableaux

3.1	Séquence globale d'adaptation.	23
3.2	Algorithme de retournement d'arêtes.	26
3.3	Algorithme de déplacement de sommets.	31
3.4	Algorithme de raffinement d'arêtes.	33
3.5	Algorithme de déraffinement de sommets.	36
3.6	Algorithme de triangulation d'une courbe Γ polygonale simple et fermée.	38
4.1	Contenu du fichier testAdaptation2DMetriqueChampScalaireUsager.txt.	49
4.2	Données des maillages pour les tests de validation.	57
4.3	Fichier de paramètres d'adaptation pour la validation.	57
4.4	Résultats pour le problème de Poiseuille : Cas $\lambda = 1.0$. Adaptation effectuée sans opération intermédiaire de nettoyage.	64
4.5	Résultats du problème de Poiseuille : Cas $\lambda = 1.0$. Adaptation effectuée avec une opération intermédiaire de nettoyage.	64
4.6	Résultats du problème de la couche limite : Cas $\lambda = 0.0$. Adaptation effectuée sans opération intermédiaire de nettoyage.	65
4.7	Résultats du problème de la couche limite : Cas $\lambda = 0.0$. Adaptation effectuée avec une opération intermédiaire de nettoyage.	65

5.1	Résultats du problème de Poiseuille : Cas $\lambda = 0.0$	88
5.2	Caractéristiques des maillages générés pour le problème de Poiseuille : Cas $\lambda = 0.0$	88
5.3	Données relatives à l'adaptation pour le problème de Poiseuille : Cas $\lambda = 0.0$.	90
5.4	Résultats du problème de Poiseuille : Cas $\lambda = 1.0$	93
5.5	Caractéristiques des maillages générés pour le problème de Poiseuille : Cas $\lambda = 1.0$	93
5.6	Données relatives à l'adaptation pour le problème de Poiseuille : Cas $\lambda = 1.0$.	94
5.7	Caractéristiques des maillages générés pour le problème de couche limite : Cas $\lambda = 0.0$	96
5.8	Résultats du problème de la couche limite : Cas $\lambda = 0.0$	97
5.9	Données relatives à l'adaptation pour le problème de couche limite : Cas $\lambda = 0.0$	98
5.10	Caractéristiques des maillages générés pour le problème de couche limite : Cas $\lambda = 1.0$	102
5.11	Résultats du problème de la couche limite : Cas $\lambda = 1.0$	103
5.12	Données relatives à l'adaptation pour le problème de couche limite : Cas $\lambda = 1.0$	104
5.13	Caractéristiques des maillages générés pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 0.0$	111
5.14	Données relatives à l'adaptation pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 0.0$	112
5.15	Caractéristiques des maillages générés pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 1.0$	117

5.16	Données relatives à l'adaptation pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 1.0$.	118
5.17	Caractéristiques des maillages générés pour le problème de la contraction : Cas $\lambda = 0.0$.	125
5.18	Données relatives à l'adaptation pour le problème de la contraction : Cas $\lambda = 0.0$.	125
5.19	Caractéristiques des maillages générés pour le problème de la contraction : Cas $\lambda = 1.0$.	132
5.20	Données relatives à l'adaptation pour le problème de la contraction : Cas $\lambda = 1.0$.	132

Introduction

De nos jours, l'ordinateur est devenu un outil incontournable de la recherche scientifique. Il ne faut pourtant pas remonter très loin pour se retrouver à une époque où les ordinateurs étaient peu nombreux, volumineux et lents. Il y avait bien sûr des "super-ordinateurs", mais ils étaient réservés à une poignée de chercheurs favorisés. C'est pourtant à ce moment-là qu'est née la simulation numérique, une technique d'analyse scientifique qui a maintenant gagné ses lettres de noblesse au même titre que la méthode expérimentale. L'apparition des micro-ordinateurs et la croissance phénoménale de leurs capacités rend cet outil de plus en plus disponible aux ingénieurs et praticiens et ce, quel que soit le type d'industrie qui les abrite.

Un des domaines dans lequel la simulation numérique a fait une percée intéressante est celui de la mécanique des fluides, en particulier, celle des fluides non-newtoniens, polymères fondus et dilués, huiles, etc., que l'on rencontre souvent en ingénierie des procédés de transformation des matériaux plastiques. Dans ces domaines, comme dans beaucoup d'autres, l'expérimentation coûte cher et la possibilité de simuler certains procédés industriels pour analyser leur validité est particulièrement attrayante.

Une des méthodes numériques de choix, maintenant couramment utilisée pour résoudre ce type de problème, est appelée Méthode des Éléments Finis (MEF). Son application à l'étude des fluides non-newtoniens fut développée vers 1975 et popularisée par des chercheurs comme P. Le Tallec, M. Fortin, M.J. Crochet et R.I. Tanner. On trouve d'ailleurs une bibliographie intéressante dans la thèse D. Esselaoui [7]. Jusqu'à il y a quelques an-

nées, seuls les problèmes bidimensionnels étaient envisageables et ce, dans des géométries simples et avec des maillages grossiers. Les principales difficultés rencontrées concernent le nombre élevé de variables qui a un impact majeur sur l'espace mémoire et la forte non-linéarité. Cette dernière influe sur le temps de calcul ainsi que sur la présence fréquente de singularités qui requiert des maillages assez fins. Les progrès initiaux furent donc lents et ce, sur tous les fronts : choix des discrétisations, méthodes de résolution, création et gestion des maillages.

Là comme ailleurs en analyse numérique, on a vu émerger des techniques dites adaptatives qui permettent de modifier le maillage de calcul dans le but de réduire la taille des systèmes linéaires à résoudre tout en améliorant la précision de la solution obtenue. Une de ces techniques, apparue dans les années 80, produit des triangulations dites de Delaunay dans lesquelles la forme des triangles dépend de la minimisation d'une fonctionnelle déterminée. Maintenant, cette technique est surtout utilisée pour la création de maillages plutôt que pour leur modification. La technique de modification la plus classique est appelée raffinement local. Dans cette technique, on cherche à déterminer les zones où le maillage n'est pas assez raffiné pour obtenir la précision visée. Une fois la zone identifiée, les triangles sont découpés en trois ou en quatre, puis des modifications sont apportées au reste du maillage (ajout d'arêtes) pour qu'il continue de respecter les règles usuelles. Cette technique peut être améliorée en introduisant le déraffinement, c'est-à-dire que l'on enlève des triangles là où il n'est pas nécessaire que leur densité soit élevée. On peut même aller jusqu'à déplacer les sommets pour régulariser les maillages.

Dans tout ceci, le point crucial est la détection des zones à modifier. Puisque le critère est la précision, il faut donc mettre au point des indicateurs d'erreur et ce, *en absence de la solution exacte*. C'est le problème de base de l'adaptation, à savoir : comment tirer de l'information de l'approximation obtenue pour modifier le maillage et obtenir une approximation de meilleure qualité ? À la base de l'étude de ces questions, on trouve les estimations *a priori* qui nous renseignent sur le lien entre la précision et le comportement

de la solution cherchée, ou plus précisément, sa régularité. La plupart de ces estimations contiennent des facteurs qui dépendent du maillage. L'idée de base du raffinement est de modifier le maillage pour diminuer ces facteurs. Plusieurs travaux dans cette direction ont été menés par Verfürth [18], Babūška [1] et Bank [2]. Dans tous les cas, on a cherché à construire des maillages pour lesquels la condition d'angle minimal, requise pour l'obtention de bonnes estimations *a priori*, soit respectée. On obtient alors des maillages dont les triangles sont de plus en plus proches d'un triangle équilatéral. De tels maillages sont dits isotropes et sont parfaitement adaptés au calcul des solutions dont la variation n'a aucune direction privilégiée.

Pourtant, en mécanique des fluides, on rencontre beaucoup de situations où les axes de variation de la solution sont dictés par la physique. C'est le cas des chocs aérodynamiques et des couches limites. Dans ce dernier cas, il y a longtemps que les praticiens ont remarqué que des triangles allongés dans la direction des lignes de niveaux de la solution donnaient une précision bien plus élevée que celle prédite par la condition d'angle. Dans ce cas, il semble donc que le maillage optimal soit un maillage orienté plutôt qu'un maillage isotrope.

Récemment, un nouveau paradigme s'est ainsi imposé, suggéré en partie par la remarque précédente mais surtout par les travaux de Simpson et D'avedezo [15]. Ces chercheurs se sont intéressés à l'adaptation de maillage du point de vue de l'interpolation. Pour résumer leurs résultats, nous énonçons d'abord précisément la question qu'ils se sont posés : étant donnée une fonction de deux variables définie sur un domaine D , quelle est la triangulation de D pour laquelle l'interpolant linéaire par morceaux de f donne la plus petite erreur ? Si f est une forme quadratique définie positive, la réponse est simple : le maillage optimal est celui pour lequel, tous les triangles sont équilatéraux, si la norme utilisée pour mesurer leurs côtés est celle définie par f . Il n'est pas difficile d'imaginer un f pour lequel un tel maillage sera fortement orienté et les triangles très étirés. Un tel maillage sera dit anisotrope. Ce résultat peut être généralisé pour une fonction f assez régulière, si on

se rappelle que l'erreur d'interpolation dépend du hessien de cette fonction f . Or, les estimations *a priori* classiques de la méthode des éléments finis disent précisément que l'erreur de discrétisation est dominée par l'erreur d'interpolation et qu'elles sont du même ordre. C'est cette remarque qui a amené M.G. Vallet à considérer l'adaptation anisotrope dans sa thèse de doctorat [17]. Cette approche est au centre des nombreux travaux récents du groupe de M. Fortin [9], membre fondateur du GIREF, un centre de recherche en analyse numérique auquel appartient l'auteur.

Dans ce mémoire, nous allons étudier ce type d'adaptation de maillages, dite anisotrope basée sur un concept de métrique non euclidienne. Par la suite, nous appliquerons cette technique à la résolution de problèmes viscoélastiques. Dans le premier chapitre, nous allons rappeler brièvement les méthodes d'éléments finis et les illustrer sur l'exemple du problème de Poisson. Dans le second chapitre, nous discuterons des estimateurs *a priori* pour ensuite introduire les estimateurs *a posteriori*. En outre, nous décrirons en détails la construction et l'utilisation de la métrique d'erreur basée sur un tel estimateur.

Dans le chapitre 3, nous détaillerons les quatre opérations élémentaires de l'adaptation à savoir le raffinement, le déraffinement, le retournement d'arêtes ainsi que le déplacement de sommets. Nous donnerons les détails des algorithmes et discuterons du bien fondé des idées principales qui soutendent ces opérations. Les travaux informatiques nécessaires pour implémenter un code d'adaptation de maillages dans un langage orienté objet tel que le C++, seront décrits dans le chapitre 4. De plus, nous validerons les opérations de base de l'adaptation sur un cas analytique. Enfin, dans le chapitre 5, nous introduirons les concepts de base de la mécanique des fluides non-newtoniens en mettant l'emphase sur le modèle le plus simple de fluide viscoélastique appelé fluide de Maxwell. Ceci fait, nous proposerons une discrétisation du modèle mathématique correspondant. Ayant choisi un estimateur *a posteriori* qui dépend d'une des composantes de la solution discrète, nous pourrions finalement l'utiliser pour tester les algorithmes d'adaptation et démontrer leur puissance sur des problèmes considérés comme très rigides.

Chapitre 1

Les méthodes adaptatives en éléments finis

Dans ce chapitre nous allons faire une brève présentation des concepts de base sur lesquels reposent ce travail. Puisque notre but est avant tout d'introduire un vocabulaire, nous ne viserons ni la profondeur ni la généralité. C'est la raison pour laquelle, bien que le problème que nous traiterons au chapitre 5 soit issu de la mécanique des fluides, nous limiterons ici notre attention au problème classique de Poisson.

1.1 Le problème modèle de Poisson

Soit Ω , un domaine régulier, fermé et connexe de \mathbb{R}^2 dont la frontière est notée Γ . On note l'opérateur Laplacien Δ par

$$\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}.$$

Soit $f(x, y)$ une fonction continue sur Ω et $g(x, y)$ une fonction continue sur Γ . On appelle solution classique du problème de Poisson, une fonction $u(x, y)$ de classe C^2 qui satisfait les conditions

$$\begin{cases} -\Delta u = f, \\ u|_{\Gamma} = g. \end{cases} \quad (1.1)$$

Sans perdre de généralité, nous choisirons $g = 0$. En effet, pour $f = f_1$ et $g = g_1$ donnés et u_1 la solution correspondante de (1.1), soit u_0 une fonction de classe C^2 quelconque pour laquelle $u_0|_{\Gamma} = g_1$. Il est facile de voir que $u = u_1 - u_0$ est une solution du même problème avec $g = 0$ et $f = f_1 + \Delta u_0$.

L'étude de l'existence et de l'unicité des solutions de (1.1) peut être menée de différentes manières, mais l'approche la plus moderne repose sur l'utilisation des idées du calcul des variations. La première étape consiste à remplacer l'équation aux dérivées partielles (EDP) par une équation variationnelle obtenue de la façon suivante. Soit ϕ une fonction régulière qui s'annule sur Γ . Multiplions les deux membres de l'EDP par ϕ puis intégrons les produits obtenus sur tout le domaine. Nous obtenons que u est la solution cherchée si

$$\int_{\Omega} -\Delta u \phi \, d\mathbf{x} = \int_{\Omega} f \phi \, d\mathbf{x} \quad \forall \phi.$$

L'application de la seconde formule de Green au membre de gauche conduit alors à

$$\int_{\Omega} \nabla u \cdot \nabla \phi \, d\mathbf{x} - \int_{\Gamma} \frac{\partial u}{\partial n} \phi \, d\mathbf{x} = \int_{\Omega} f \phi \, d\mathbf{x} \quad \forall \phi \quad (1.2)$$

où n dénote la normale unitaire extérieure et $\frac{\partial u}{\partial n} = n \cdot \nabla u$ la dérivée normale de u .

Ce calcul est purement formel, puisque nous n'avons ni précisé la nature des fonctions ϕ ni vérifié la validité des différentes intégrations. Cette étape est cruciale, mais nous ne la développerons pas. Pour de plus amples détails, le lecteur peut consulter [5] et [12]. Notons tout de même que, pour que les intégrales apparaissant dans (1.2) aient un sens, il suffit que $\nabla u, \phi, f$ ainsi que leurs produits soient intégrables. Nous désignerons donc par

$$\mathcal{V} := H_0^1(\Omega) = \{v \in L^2(\Omega) \mid \nabla v \in (L^2(\Omega))^2, \quad v|_{\Gamma} = 0\},$$

l'espace des fonctions de L^2 dont les dérivées partielles au sens des distributions sont aussi dans L^2 . Ensuite, nous chercherons la solution de (1.2) dans cet espace. Le problème se

réduit alors à un problème assez simple d'inversion d'un opérateur auto-adjoint défini positif dans un espace de Hilbert (voir [12]). Même si nous pouvons montrer que (1.2) possède une solution unique, ce ne sera pas tout. Il restera à montrer que cette solution est une solution du problème de départ. De tels résultats sont appelés résultats de régularité. Ils sont généralement difficiles à démontrer, mais ils en existent plusieurs. Toute solution de (1.2) est appelée solution faible de (1.1). Dans le cas où la frontière Γ n'a pas de point anguleux trop prononcé, on peut montrer qu'une solution faible est aussi une solution classique. Dans le cas contraire, il se peut qu'il n'existe que des solutions faibles. Ainsi, ce que nous chercherons à calculer sera la solution faible et ce, dans tous les cas.

Puisqu'il existe assez peu de cas où on peut obtenir une expression analytique de la solution de (1.1), nous sommes naturellement conduit aux méthodes numériques dites de discrétisation. Parmi celles-ci, nous favoriserons la méthode des éléments finis (MEF).

1.2 Discrétisation du problème de Poisson

Pour résoudre numériquement le problème (1.2), nous adoptons une idée naturelle due à Galerkin. Donnée un espace de dimension finie $\mathcal{W} \subseteq \mathcal{V}$, on cherche une approximation $\tilde{u} \in \mathcal{W}$ de u qui est solution du problème :

$$\int_{\Omega} \nabla \tilde{u} \cdot \nabla w \, d\mathbf{x} = \int_{\Omega} f w \, d\mathbf{x} \quad w \in \mathcal{W}. \quad (1.3)$$

Si on connaît une base $\mathcal{B} = \{w_1, \dots, w_n\}$ de \mathcal{W} , le système précédent se réduit à un nombre fini d'équations

$$\int_{\Omega} \nabla \tilde{u} \cdot \nabla w_i \, d\mathbf{x} = \int_{\Omega} f w_i \, d\mathbf{x} \quad \forall w_i, \quad i = 1, \dots, n.$$

En exprimant \tilde{u} dans la base \mathcal{B} par $\tilde{u} = \sum_1^n \tilde{u}_i w_i$, et en substituant dans l'équation précédente, le problème est ramené à celui de résoudre le système linéaire

$$\mathbf{A} \begin{pmatrix} \tilde{u}_1 \\ \vdots \\ \tilde{u}_n \end{pmatrix} = \mathbf{B} \quad (1.4)$$

dans lequel la matrice $\mathbf{A} = (a_{i,j})$ et le vecteur $\mathbf{B} = (b_i)$ sont définis par

$$a_{i,j} = \int_{\Omega} \nabla w_i \cdot \nabla w_j \, d\mathbf{x}, \quad \text{et } b_i = \int_{\Omega} f w_i \, d\mathbf{x}.$$

Comme il faut s'y attendre, la précision de la méthode va croître avec la dimension de \mathcal{W} , c'est-à-dire celle du système linéaire. Pour que cette méthode soit viable, il faut donc choisir des bases pour lesquelles le système (1.4) est facile à résoudre, bien qu'il soit de grande dimension.

Pour des domaines simples, on peut choisir des bases orthogonales au sens du produit scalaire défini par \mathbf{A} et obtenir un système diagonal. Cette approche, dite spectrale, n'est cependant pas applicable aux domaines complexes. On doit alors utiliser une autre méthode qui donnera une matrice creuse. Cette méthode est appelée méthode des éléments finis. Nous allons en donner une description dans le cas où Ω est un polygone (pour le cas général, voir [5]).

On se donne d'abord une triangulation de Ω , c'est-à-dire une partition

$$\mathcal{T}_h = \{T_i | i = 1, \dots, n_e\}$$

où T_i est un triangle droit, n_e est le nombre d'éléments triangulaires et tel que \mathcal{T}_h satisfait les conditions suivantes

1. $\Omega = \bigcup_{i=1}^{n_e} T_i$;
2. si $i \neq j$, $T_i \cap T_j$ coïncide avec
 - (a) l'ensemble vide,
 - (b) un sommet commun aux deux triangles,
 - (c) une arête commune aux deux triangles.

Avec cette triangulation, on peut maintenant construire le sous-espace de \mathcal{V} des fonctions polynômiales par morceaux de degré donné

$$\mathcal{W}_h = \{w_h \in C^0(\Omega) \mid w_h|_{T_i} \in P_k(T_i), \, i = 1, \dots, n_e\}$$

où $P_k(T_i)$ est l'espace des polynômes de degré inférieur ou égal à k défini sur T_i . C'est à partir de cette suite d'espaces que nous allons appliquer la méthode de Galerkin. Il ressort de la définition de \mathcal{W}_h que cette discrétisation est entièrement définie par deux paramètres :

1. la triangulation \mathcal{T}_h ,
2. le degré k de l'approximation des polynômes.

Les éléments de \mathcal{T}_h sont appelés éléments finis (EF), d'où le nom de la méthode. Le diamètre de $T \in \mathcal{T}_h$ est noté h_T ,

$$h_T = \max_{(x,y) \in T} \|x - y\|_{\mathbb{R}^2}.$$

La "taille du maillage" sera mesurée par

$$h = \max_{T \in \mathcal{T}_h} h_T.$$

De même, la "rotondité" de T , c'est-à-dire le diamètre du cercle inscrit sera notée ρ_T . Un maillage sera dit régulier s'il existe une constante σ indépendante de h pour laquelle

$$\max_{T \in \mathcal{T}_h} \frac{h_T}{\rho_T} \leq \sigma. \quad (1.5)$$

Nous aurons aussi besoin de considérer des sous-ensembles de la triangulation \mathcal{T}_h . On appellera **coquille** un sous-domaine fermé connexe du maillage global. On construira une coquille à partir d'une entité du maillage (e.g. une arête, un sommet) en incluant tous les éléments qui sont adjacents à cette entité. La figure 1.1 est un exemple d'une coquille extraite autour d'un sommet S .

Pour revenir à la méthode de Galerkin, nous devons maintenant construire une base. Comme nous l'avons vu, le choix de cette base doit être tel que la matrice \mathbf{A} qui en résultera sera creuse, c'est-à-dire qu'un grand nombre des intégrales $\int_{\Omega} \nabla w_i \cdot \nabla w_j \, dx$ seront nulles.

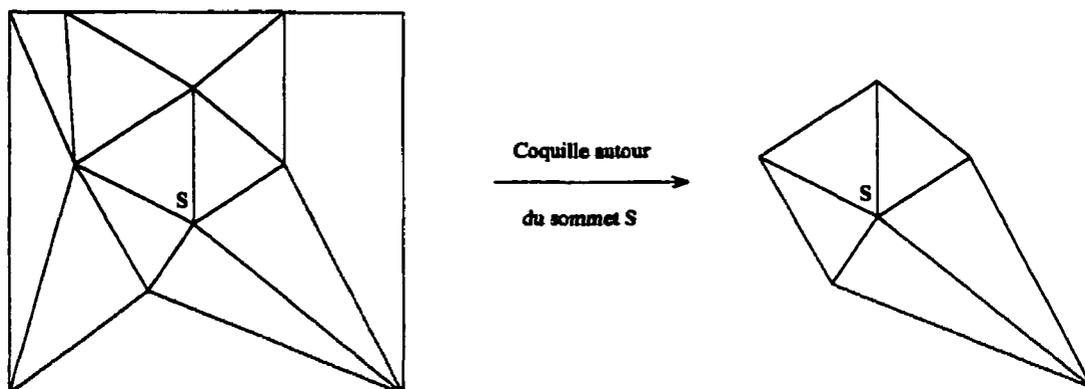


FIG. 1.1 – Coquille extraite d'un maillage.

Nous allons décrire cette construction dans le cas des polynômes de degré un ($k = 1$), pour les autres cas, le lecteur peut consulter [12]. Notons s_1, \dots, s_{n_s} les n_s sommets de la triangulation \mathcal{T}_h . À chaque sommet s_i , nous associons une fonction linéaire par morceaux w_i qui vaut 1 en s_i et 0 aux autres sommets. Le support de cette fonction (i.e. la fermeture de l'ensemble des points de Ω où elle est non nulle) n'est rien d'autre que la "coquille" associée à ce sommet. Puisque, pour de nombreuses paires de sommets, l'intersection des coquilles est vide, il en résulte immédiatement que la matrice sera creuse.

En outre, pour w_h une fonction de \mathcal{W}_h , on la "formule de Lagrange" :

$$w_h(x, y) = \sum_{i=1}^{n_s} w_h(s_i) w_i(x, y). \quad (1.6)$$

C'est donc dire que le vecteur des coefficients de w_h dans cette base, appelés degrés de liberté (ddl), sont les valeurs de w_h aux sommets (valeurs nodales). Ainsi, la solution du système (1.4) donnera les valeurs nodales de \tilde{u} et \tilde{u} pourra être calculée en tout point par (1.6).

Au chapitre suivant, nous verrons que la qualité de l'approximation de u par \tilde{u} dépendra effectivement du maillage et du degré des polynômes. Cette qualité pourrait donc être améliorée de deux façons. La première approche, que les ingénieurs appellent la "p-method", consiste à fixer le maillage et à changer k . Dans le cas d'un seul élément, elle se

réduit à l'approche spectrale. La deuxième consiste à fixer k et à modifier le maillage. Les ingénieurs appellent cette approche "h-method" et c'est celle-ci que nous favoriserons.

1.3 Adaptation

Dans notre contexte, l'idée d'adaptation, qui est très répandue en analyse numérique, peut se résumer comme suit : donnée(s) une ou des approximations de u , peut-on obtenir de l'information à partir de laquelle on pourrait calculer un maillage sur lequel la nouvelle approximation serait meilleure ?

Considérons un exemple simple. Pour calculer $I = \int_0^1 f(x) dx$ on peut utiliser les approximations du trapèze (T) et de Simpson(S). On obtient ainsi deux approximations I_T et I_S . Si l'écart $\|I_T - I_S\|$ est petit, il est raisonnable de considérer que I_T est une bonne approximation. Sinon, on peut augmenter le nombre de noeuds d'intégration et recommencer.

Ce raisonnement est basé sur le fait que la quantité $\|I_T - I_S\|$ est un **bon estimateur** de l'erreur $\|I - I_T\|$ commise par l'approximation du trapèze. Pour transposer cette idée dans notre contexte, nous devons donc :

1. Choisir un estimateur.
2. Décider comment "améliorer" les maillages.

Nous discuterons du premier choix au chapitre 2. Le problème de l'amélioration des maillages est au centre de ce travail et sera abordé au chapitre 3.

Chapitre 2

Estimation d'erreur

Dans ce second chapitre, nous nous concentrerons sur la construction d'estimateurs d'erreur ce qui n'est évidemment possible que si nous avons une bonne définition de l'erreur elle-même. Par exemple, si nous revenons au problème de Poisson, la qualité de la discrétisation sera mesurée par $\|u - u_h\|$ où u_h est la solution du problème (1.3), u est celle de (1.2) et $\| \cdot \|$ est une norme à déterminer. Notons que le choix de la norme influencera le résultat. Nous voulons choisir une norme qui est naturelle et facile à calculer, c'est pourquoi nous nous limiterons à la norme la plus simple qui est la norme de \mathcal{V} .

Il convient ici d'introduire quelques notations. Notons que \mathcal{V} est un espace de Sobolev, c'est-à-dire un sous-espace de L^2 de fonctions dérivables. Si les fonctions sont ℓ fois dérivables le sous-espace est noté $H^\ell(\Omega)$. Dans le cas où $\ell = 0$, l'espace est L^2 muni de la norme usuelle

$$|v|_0 = \sqrt{\left(\int_{\Omega} v^2 d\mathbf{x}\right)^2}. \quad (2.1)$$

Pour $\ell > 0$, on peut doter chacun des sous-espaces d'une semi-norme

$$|v|_{\ell} = \sqrt{\sum_{|\alpha|=\ell} \left| \frac{\partial^{|\alpha|} v}{\partial x^{\alpha}} \right|_0^2} \quad (2.2)$$

(où nous utilisons la notation classique des multi-indices) et de la norme correspondante

$$\|v\|_\ell = \sqrt{\sum_{k=1}^{\ell} |v|_k^2}. \quad (2.3)$$

Dans ce travail, nous nous intéresserons surtout aux cas $\ell = 0, 1, 2$ et nous mesurerons la qualité de l'approximation par la norme associée.

2.1 Estimation d'erreur *a priori*

Dans cette section, nous discuterons brièvement des résultats théoriques connus sur l'erreur de discrétisation. Le lecteur peut consulter [5] ou [12] pour connaître les démonstrations des différents théorèmes qui seront énoncés.

Le premier théorème que nous considérons est un résultat général qui s'applique à la discrétisation d'une vaste classe de problèmes variationnels.

Théorème 1 *Il existe une constante $C > 0$, indépendante du sous-espace \mathcal{V}_h de \mathcal{V} , telle que*

$$\|u - u_h\| \leq C \inf_{v_h \in \mathcal{V}_h} \|u - v_h\|. \quad (2.4)$$

On déduit de ce résultat que l'erreur de discrétisation est toujours du même ordre que la meilleure approximation de u dans \mathcal{V}_h . Ceci nous fournira une façon simple d'obtenir un estimateur de $\|u - u_h\|$.

Si on considère u une fonction définie partout et qu'on désigne par $\pi_h u$ l'interpolé de u dans \mathcal{V}_h qui s'écrit comme

$$\pi_h u = \sum_{i=1}^{n_s} u_i v_i \quad v_i \in \mathcal{V}_h, u_i \in \mathbb{R},$$

en appliquant le théorème 1, on obtient le résultat suivant :

$$\|u - u_h\| \leq C \|u - \pi_h u\|.$$

L'intérêt de ce résultat réside dans la simplicité du majorant et dans le fait que la théorie de l'approximation nous en fournit une estimation optimale. Plus précisément, on en déduit le théorème suivant.

Théorème 2 *Soit \mathcal{T}_h une triangulation donnée et \mathcal{V}_h un espace de fonctions polynômiales par morceaux, continues et de degré $k \geq 1$ sur chaque triangle. Alors, il existe une constante C qui ne dépend que de Ω et k telle que pour tout entier m , $0 \leq m \leq k+1$ et pour tout $K \in \mathcal{T}_h$,*

$$|v - \pi_h v|_{m,K} \leq C \frac{h_K^{k+1}}{\rho_K^m} |v|_{k+1,K} \quad \forall v \in H^{k+1}(K). \quad (2.5)$$

Une telle estimation, valide sur un seul élément, est appelée locale. Notons qu'on peut passer d'une estimation locale à une estimation globale en sommant sur tous les éléments.

Ainsi, en posant $m = 1$ et $u \in H^{k+1}(\Omega)$ dans le théorème 2 et en sommant sur tous les triangles K de \mathcal{T}_h , on obtient l'estimation globale :

$$|u - \pi_h u|_{1,\Omega} \leq C h^k \sum_{i=1}^{n_e} \left(\frac{h_K}{\rho_K} \right) |u|_{k+1,K}. \quad (2.6)$$

On rencontre alors une difficulté liée à la quantité géométrique $\frac{h_K}{\rho_K}$ qui est une mesure de l'aplatissement de l'élément. Sous l'hypothèse que la triangulation \mathcal{T}_h est régulière (voir (1.5)), nous pouvons cependant déduire que :

$$|u - \pi_h u|_{1,\Omega} \leq C \sigma h^k |u|_{k+1,\Omega}. \quad (2.7)$$

Dans le cas $k = 1$, on remarque que le majorant contient le terme

$$|u|_{2,\Omega} = \left(\sum_{i,j=1}^2 \left| \frac{\partial^2 u}{\partial x_i \partial x_j} \right|_{0,\Omega} \right)^{\frac{1}{2}}.$$

Cette quantité jouera un rôle important par la suite.

2.2 Estimation d'erreur *a posteriori*

Cette section présente les méthodes d'adaptation de maillages inspirées de la librairie LIBOM. Cette librairie a été conçue et développée dans le cadre de la thèse de doctorat de Mme Vallet [17]. Par la suite, elle a été largement développée par l'équipe de M. Fortin et W. Habashi [9]. Ainsi, nous aborderons des notions fondamentales telles les estimateurs *a posteriori*, les métriques d'erreurs, les fonctions de raffinement, la qualité des éléments et l'erreur cible.

2.2.1 Estimateurs d'erreur *a posteriori*

Dans cette sous-section, nous voulons faire le lien entre l'estimateur *a priori* de la section précédente, qui n'est pas effectivement calculable, et un estimateur *a posteriori* dont l'évaluation ne dépend que de la solution discrète u_h . L'idée qui sous-tend la construction de l'estimateur est simple, mais la démonstration de sa validité l'est moins.

En bref, nous voulons remplacer le facteur qui dépend de u dans le membre de droite de (2.7) par une quantité évaluée à partir de u_h . A priori, cette idée n'a aucun sens puisque u_h est linéaire et que les dérivées deuxièmes sont nulles. Cependant, il faut tenir compte des sauts des dérivées premières aux arêtes. Ce sont ces sauts qui nous permettent d'obtenir une approximation valable des dérivées deuxièmes.

La démonstration de la validité d'une telle idée n'est pas à notre portée. On peut consulter [17] pour un exemple d'approche possible. Nous allons présenter ici que la méthode de calcul qui est encore basée sur une formulation variationnelle. Plus précisément, supposons que nous voulions calculer $\frac{\partial^2 u_h}{\partial x_i \partial x_j}$ et écrivons

$$H_{i,j} = \frac{\partial^2 u_h}{\partial x_i \partial x_j} \iff \int_{\Omega} H_{i,j} v \, d\mathbf{x} = \int_{\Omega} \frac{\partial^2 u_h}{\partial x_i \partial x_j} v \, d\mathbf{x}, \quad \forall v \in H_0^1(\Omega).$$

Intégrons maintenant le membre de droite par parties pour obtenir

$$\int_{\Omega} H_{i,j} v \, d\mathbf{x} = - \int_{\Omega} \frac{\partial u_h}{\partial x_i} \frac{\partial v}{\partial x_j} \, d\mathbf{x}.$$

Donc, chacune des entrées de la matrice Hessienne \mathbf{H} peut être obtenue par la résolution d'un système linéaire de même dimension que le système de départ. Malgré la grande taille de la matrice, celle-ci est bien conditionnée et le système peut être résolu rapidement à l'aide d'une méthode de gradient conjugué préconditionné par SSOR (voir [14] pour de plus amples détails sur ces méthodes). En pratique cette idée est surtout utilisée pour des problèmes plus complexes que celui étudié ici, qui comportent souvent plusieurs variables parmi lesquelles il faut choisir. En outre, cette approche ne permet d'obtenir le Hessien que sur les noeuds intérieurs. Il faudra encore le corriger sur le bord. Dans ce cas, les seules inconnues sont les valeurs au bord ce qui amène à construire un système linéaire de faible dimension. Dans la section suivante, nous présentons un résumé de toutes ces idées sous forme algorithmique.

2.2.2 Approximation des dérivées du deuxième ordre d'une fonction linéaire par morceaux

Données : S une fonction linéaire par morceaux dont on veut approximer le Hessien $\mathbf{H} = \text{Hessien désiré} = \{\mathbf{H}_{i,j}\}_{i,j=1}^2$. Chaque $\mathbf{H}_{i,j}$ est une fonction linéaire par morceaux définie en chaque sommet.

Initialisation : $\mathbf{H} = \mathbf{0}$ et S donné. Posons $n_{\mathcal{V}_h}$ le nombre de fonctions de base dans l'espace de discrétisation \mathcal{V}_h .

Étape 1 : Calcul du Hessien sur l'intérieur de $\Omega \setminus \Gamma$:

Pour chaque composante $\mathbf{H}_{i,j}$ du tenseur \mathbf{H} on résout

$$\mathbf{M}\mathbf{H}_{i,j} = \mathbf{B}_{i,j}$$

où la matrice masse $\mathbf{M} = \{M_{k,l}\}_{k,l=1}^{n_{v_h}}$ et le vecteur $\mathbf{B}_{i,j} = \{(\mathbf{B}_{i,j})_k\}_{k=1}^{n_{v_h}}$ sont donnés par

$$M_{k,l} = \int_{\Omega} v_k v_l \, dx$$

$$(\mathbf{B}_{i,j})_k = - \int_{\Omega} \frac{\partial S}{\partial x_i} \frac{\partial v_k}{\partial x_j} \, dx.$$

Étape 2 : Calcul du Hessien sur le bord Γ :

Pour chaque composante $\mathbf{H}_{i,j}$ de la matrice \mathbf{H} on résout

$$\begin{cases} \mathbf{A}\mathbf{x} = \mathbf{B}_{i,j} \\ \mathbf{x}|_{\Omega \setminus \Gamma} = 0 \end{cases}$$

où la matrice laplacienne $\mathbf{A} = \{a_{k,l}\}_{k,l=1}^{n_{v_h}}$, le vecteur $\mathbf{B}_{i,j} = \{(\mathbf{B}_{i,j})_k\}_{k=1}^{n_{v_h}}$ et $\{v_k\}_{k=1}^{n_{v_h}} = \mathbf{V}_h$ sont donnés par

$$a_{k,l} = \int_{\Omega} \nabla v_k \cdot \nabla v_l \, dx$$

$$\mathbf{B}_{i,j} = -\mathbf{A}\mathbf{H}_{i,j}.$$

On corrige ensuite le Hessien sur le bord

$$\mathbf{H}_{i,j} \leftarrow \mathbf{H}_{i,j} + \mathbf{x}.$$

Étape 3 : Normalisation des Hessiens :

Le Hessien ainsi obtenu servira à définir une nouvelle métrique sur Ω . Ceci n'est bien sûr possible que si la matrice correspondante est définie positive. Si elle ne l'est pas, nous utiliserons la diagonalisation de cette matrice pour la remplacer par une matrice définie positive qui partage les mêmes axes principaux. Pour éviter les élongations excessives, nous voulons aussi maintenir les valeurs propres dans un intervalle $[v_{min}, v_{max}]$. Pour ce faire, nous procédons comme suit.

Pour chaque valeur \mathbf{t}_{s_i} du champ tensoriel \mathbf{H} au sommet s_i ($i = 1, \dots, n_s$), on calcule les deux valeurs propres v_1, v_2 et les deux vecteurs propres associés $\mathbf{V}_1, \mathbf{V}_2$. Le tenseur \mathbf{t}_{s_i} s'écrit donc

$$\mathbf{t}_{s_i} = \begin{pmatrix} \mathbf{V}_1 & \mathbf{V}_2 \end{pmatrix} \begin{pmatrix} v_1 & 0 \\ 0 & v_2 \end{pmatrix} \begin{pmatrix} \mathbf{V}_1 & \mathbf{V}_2 \end{pmatrix}^T. \quad (2.8)$$

On remplace les valeurs propres v_i par des quantités positives de l'intervalle désiré

$$\text{Pour } i = 1, 2 \quad \begin{cases} 1. v_i \leftarrow \|v_i\| \\ 2. v_i \leftarrow \max(v_i, v_{\min}) \\ 3. v_i \leftarrow \min(v_i, v_{\max}). \end{cases}$$

On reconstruit le tenseur avec les nouvelles valeurs propres $\{v_i\}_{i=1}^2$ et les mêmes vecteurs propres $\{\mathbf{V}_i\}_{i=1}^2$ en utilisant la formule (2.8).

Cette construction nous permet d'obtenir un champ de tenseurs, linéaire par morceaux, défini à chacun des sommets du maillage. À partir de ce champ, nous pourrions évaluer le membre de droite de (2.5) pour estimer l'erreur sur un triangle donné. Si cette erreur est trop grande, on pourrait diviser le triangle et ainsi, de triangle en triangle, obtenir un nouveau maillage. Une telle approche, qui ne tient compte que de la taille de l'erreur et non de la direction de sa variation maximale, est dite isotrope. Nous ne retiendrons pas cette approche.

2.2.3 Fonction de raffinement

Une fonction de raffinement est une fonction $h : \Omega \rightarrow \mathbb{R}^+$ qui détermine la taille désirée des éléments en fonction de leur position dans Ω . Par exemple, la valeur de $h(\mathbf{x})$ peut représenter la longueur minimale ou maximale d'une arête passant par le point $\mathbf{x} \in \Omega$ ou le diamètre d'un élément qui contient \mathbf{x} .

Dans ce présent travail, la fonction de raffinement sera basée sur la métrique construite à partir du Hessien calculé à la section 2.2.2. De plus, $h(\mathbf{x})$ sera la longueur désirée de l'arête passant par $\mathbf{x} \in \Omega$.

2.2.4 Raffinement anisotrope

Définir une bonne fonction de raffinement, c'est construire un maillage pour lequel l'estimateur d'erreur *a posteriori* représente bien l'erreur d'interpolation. Des travaux récents de Simpson et Davedezo [15] suggèrent que le maillage optimal est celui pour lequel toutes les arêtes ont la même longueur lorsque celle-ci est calculée non pas dans la métrique euclidienne, mais plutôt dans la métrique définie par le Hessien de u . C'est de cette manière que nous utiliserons (2.5).

Pour appliquer ce critère, il faut avoir une méthode pour évaluer la longueur des arêtes. Nous la présenterons à la sous-section 2.2.5. Cependant, on ne peut guère s'attendre à atteindre une valeur commune à toutes les arêtes et il faut permettre une certaine variation sur la longueur cible. Cette tolérance sera décrite dans la sous-section 2.2.6.

Cette idée d'uniformité des longueurs d'arêtes est une des idées maîtresses de **LIBOM**. Ainsi, on répartit uniformément l'erreur sur chaque arête du maillage. Or, le fait d'utiliser une métrique non-euclidienne crée des arêtes de longueur euclidienne différente. On a donc des triangles déformés, i.e. allongés dans une direction particulière. On qualifie d'**anisotrope** un maillage avec de tels éléments. Ce maillage a donc des directions privilégiées où l'erreur est plus importante et où la concentration des éléments sera plus grande.

Par contre, si la métrique est euclidienne, les triangles auront leurs arêtes sensiblement de la même longueur et aucune direction ne sera privilégiée. Un tel maillage est dit **isotrope**.

2.2.5 Calcul de la longueur d'une arête dans la métrique

Pour calculer la longueur d'une arête, disons $\overline{s_1 s_2}$, on utilise les tenseurs \mathbf{H}_{s_1} et \mathbf{H}_{s_2} qui sont les tenseurs de la métrique définis aux sommets $s_1 = (x_1, y_1)$ et $s_2 = (x_2, y_2)$ respectivement. Puisque l'approximation est linéaire, on peut évaluer le tenseur au milieu de l'arête en prenant la moyenne. Posons $\mathbf{H}_\mu = \frac{\mathbf{H}_{s_1} + \mathbf{H}_{s_2}}{2}$ et $a = (x_2 - x_1, y_2 - y_1)$ le vecteur représentant l'arête $\overline{s_1 s_2}$.

La longueur $l_{\overline{s_1 s_2}}$ est calculée de la façon suivante

$$l_{\overline{s_1 s_2}} = \sqrt{a \mathbf{H}_\mu a^t}.$$

2.2.6 Erreur cible et tolérance

Si on note par a_i chacune des n_a arêtes de la triangulation \mathcal{T}_h , par $l_{a_i}^m$ la longueur de l'arête a_i dans la métrique et par l_d^m la longueur désirée des arêtes dans la métrique, alors on définit

$$\mu_{\mathcal{T}_h} = \frac{1}{n_a} \sum_{i=1}^{n_a} l_{a_i}^m$$

$$\sigma_{\mathcal{T}_h} = \left(\frac{1}{n_a} \sum_{i=1}^{n_a} (l_{a_i}^m - \mu_{\mathcal{T}_h})^2 \right)^{\frac{1}{2}} = \left(\frac{1}{n_a} \sum_{i=1}^{n_a} (l_{a_i}^m)^2 - \mu_{\mathcal{T}_h}^2 \right)^{\frac{1}{2}}$$

où $\mu_{\mathcal{T}_h}$ représente la moyenne des longueurs $l_{a_i}^m$ de la triangulation \mathcal{T}_h et $\sigma_{\mathcal{T}_h}$ l'écart type des $l_{a_i}^m$.

On pose n_{div} , le nombre de divisions désirées des arêtes les plus longues. On utilisera $n_{div} = 4$ pour les tests numériques. Dans la métrique de l'erreur, on détermine l_c^m la longueur cible désirée pour chaque arête du maillage par l'équation suivante

$$l_c^m = \max \left(l_d^m, \frac{\mu_{\mathcal{T}_h} + \sigma_{\mathcal{T}_h}}{n_{div}} \right). \quad (2.9)$$

Maintenant que la longueur cible est déterminée, on peut définir la fonction de raffinement telle que décrite à la section 2.2.3. On prendra $h(\mathbf{x})$ comme la longueur que doivent avoir les arêtes passant par \mathbf{x} . En fait, $h(\mathbf{x})$ est constante sur tout le domaine et vaut $h(\mathbf{x}) = l_c^m$.

On doit aussi se donner une tolérance sur la longueur permise des arêtes dans le maillage adapté. Pour se faire, on utilisera le facteur de tolérance δ_l et on permettra qu'une arête a ait une longueur l_a^m comprise dans l'intervalle $\left[\frac{l_c^m}{\delta_l}, l_c^m \cdot \delta_l\right]$. De plus, on définit le critère d'adaptation comme suit :

on doit adapter si la longueur l_a^m ne satisfait pas l'inégalité

$$\frac{l_c^m}{\delta_l} \leq l_a^m \leq l_c^m \cdot \delta_l. \quad (2.10)$$

Par exemple, on peut choisir $\delta_l = 1.5$ comme tolérance. On ne doit pas descendre en-dessous de $\sqrt{2}$ car certaines opérations géométriques nécessaires peuvent changer la longueur l_a^m par ce facteur.

De plus, pour prévenir la création d'arêtes de longueur euclidienne l_a^e trop petites dans le cas où l'erreur est infinie ou bien pour empêcher d'avoir des arêtes de longueur l_a^e trop grande, on impose que

$$l_{min}^e \leq l_a^e \leq l_{max}^e. \quad (2.11)$$

On doit donc s'assurer que toutes les arêtes du maillage satisfont (2.10) et (2.11).

2.2.7 Qualité des éléments

Généralement, l'utilisation d'une métrique conduit à la génération de maillages anisotropes. Avec ce type de maillages, on ne peut se permettre d'avoir des triangles trop étirés, car la convergence numérique peut être perturbée. Pour éviter ce problème, on doit donc construire des maillages dont les triangles sont les moins déformés possible. Dans le cas

d'une métrique euclidienne, les triangles sont peu déformés et tendent vers l'optimum qui est le triangle équilatéral.

Pour reprendre cette idée dans le cas anisotrope, il faut se donner un critère pour quantifier l'équilatéralité d'un triangle. Pour ce faire, on définit la qualité Q_K d'un triangle K comme étant une valeur comprise dans l'intervalle $[0, 1]$ telle que $Q_K = 0$ représente un triangle aplati et $Q_K = 1$, un triangle équilatéral. On se base sur la formule d'Héron pour évaluer Q_K

$$Q_K = 27 \frac{(p-a)(p-b)(p-c)}{p^3} \quad (2.12)$$

avec a, b, c les longueurs des côtés de K et p le demi-périmètre du triangle K .

2.2.8 Quelques remarques

On peut se demander pourquoi nous avons choisi l'approche anisotrope. La réponse tient de l'analyse des coûts. Dans le cas où la solution d'un problème varie rapidement sur une bande très mince, ce qui est le cas dans la simulation des chocs aérodynamiques, l'approche isotrope conduira à un maillage de la bande composé d'un très grand nombre de triangles de petit diamètre. Par contre, l'approche anisotrope permettra de mailler la même bande avec peu de triangles très allongés. Ceci se traduit bien sûr par un gain de mémoire et de temps de calcul.

Cependant, cette approche crée aussi ses problèmes. Ainsi, dans le cas instationnaire, le choc peut se déplacer dans le temps et un maillage qui était parfaitement adapté au temps t peut être très inapproprié au temps $t + \Delta t$, au point même de conduire à la divergence du résolveur. Il est donc parfois utile de redémarrer sur un maillage isotrope grossier.

Chapitre 3

L'adaptation de maillages

L'adaptation de maillages est une des étapes de la séquence globale d'adaptation définie au tableau 3.1 ci-dessous et que l'on doit parcourir dans le cadre de la résolution des problèmes variationnels par la méthode des éléments finis. l_d^m et l_c^m sont définis à la section (2.2.6).

<p>l_d^m donné.</p> <p>Poser $l_c^m = l_d^m + 1$.</p> <p>Tant que $l_c^m > l_d^m$, faire :</p> <p> Calculer la solution u_h du problème variationnel sur le maillage courant.</p> <p> Utiliser la solution discrète u_h trouvée et calculer la métrique M de l'erreur.</p> <p> Réf. section 2.2.2.</p> <p> Calculer l_c^m à partir de M et de (2.9).</p> <p> Si $l_c^m \leq l_d^m$ arrêter.</p> <p> ADAPTATION : Adapter avec la métrique M calculée pour obtenir un nouveau maillage \mathcal{T}_h. On exécute la séquence définie par l'équation (3.2).</p> <p> Remplacer le maillage courant par le maillage \mathcal{T}_h.</p> <p>Fin.</p>
--

TAB. 3.1 – Séquence globale d'adaptation.

Ici, nous voulons nous concentrer sur l'étape ADAPTATION de cette séquence globale. Cette étape comporte une suite d'opérations de base qui sont appliquées au maillage courant pour minimiser l'erreur de discrétisation. Nous allons nous baser sur la **LIBrairie d'Optimisation de Maillages (LIBOM)** et utiliser ses quatre opérations de base, à savoir : le raffinement, le déraffinement, le retournement d'arêtes ainsi que le déplacement de sommets. La prise de décision quant à l'utilisation de l'une ou l'autre de ces opérations reposera sur le paradigme fondamental de LIBOM qui veut que l'erreur commise soit, en quelque sorte, représentée par la longueur des arêtes du maillage et que le maillage optimal soit celui pour lequel chaque arête est de la même longueur lorsqu'elle est mesurée dans la métrique de l'erreur. Notre but, dans ce chapitre, est de proposer une description de la base théorique et du fonctionnement de ces quatre opérations.

Comme nous l'avons dit, le but visé est de satisfaire le critère d'uniformisation $h(\mathbf{x}) = Cte$ (sec. 2.2.6), où la fonction h dépend de la métrique calculée à partir du Hessien (Sec. 2.2.2). Comme une modification globale du maillage est difficilement envisageable, nous travaillerons plutôt arête par arête ou sommet par sommet en étudiant l'impact d'une opération sur la coquille associée à l'entité plutôt que sur le maillage global.

Le raffinement d'arêtes permet de couper les arêtes trop longues dans la métrique tandis que le déraffinement d'arêtes permet d'enlever les arêtes trop courtes. Ces deux opérations ont un effet marqué sur le nombre de degrés de liberté (ddls), car elles forcent l'ajout ou le retrait de sommets. En outre, elles affectent fortement la topologie. C'est pourquoi elles doivent être suivies d'une régularisation qui ne modifiera pas le nombre de ddls. Il existe deux opérations standards pour uniformiser la longueur des arêtes sans ajouter d'élément tout en augmentant la qualité Q_K (2.12) des éléments : le déplacement de sommets et le retournement d'arêtes. Le déplacement de sommets utilise la métrique de l'erreur pour uniformiser la longueur des arêtes. Par ailleurs, le retournement est d'abord de nature géométrique et utilise peu la métrique.

Après avoir décrit les quatre opérations, nous discuterons, à la section 3.5, de leur ordre d'appel.

3.1 Retournement d'arêtes

Une façon simple et efficace de rendre le maillage plus régulier consiste à changer l'orientation de certaines arêtes. Cette opération est d'abord et avant tout géométrique, puisque la qualité Q_K (2.12) des éléments est basée sur la formule de Héron dans laquelle n'interviennent que les longueurs euclidiennes des côtés du triangle. Ainsi, pour chaque arête du maillage, on examinera la possibilité d'améliorer la qualité des éléments adjacents en utilisant l'algorithme décrit dans le tableau 3.2 ci-après.

L'effet sur le maillage est très prononcé et le maillage résultant possède moins de triangles aplatis. Les deux cas typiques de retournement (retournement d'une arête intérieure ou du bord) seront respectivement détaillés aux sections 3.1.1 et 3.1.2 et illustrés aux figures 3.1 et 3.2.

Comme pour toutes les autres opérations, nous devrions prendre la précaution de ne pas créer d'arêtes trop longues ou trop courtes en utilisant les paramètres l_{min}^e et l_{max}^e . Cependant, la discussion présentée à la section 4.4.5 montre qu'il n'est pas toujours désirable de le faire car cela pourrait empêcher certains retournements et diminuer grandement la qualité des éléments. En fait, notre position est que le retournement d'arêtes doit d'abord et avant tout améliorer la qualité Q_K des éléments.

3.1.1 Retournement d'une arête interne

On dira que l'arête est interne au maillage ou à la coquille si cette arête est adjacente à deux triangles. Dans la figure 3.1, l'arête formée par les sommets S1 et S2 est interne.

Pour chaque arête a du maillage global, faire :

Si le retournement d'arêtes n'est pas possible ou qu'il crée des éléments superposés, passer à l'arête suivante.

Construire une coquille autour de l'arête a .

Calculer la qualité minimale des éléments de la coquille.

Retourner l'arête dans la coquille comme suit :

Si l'arête est interne, retourner l'arête tel que décrit à la section 3.1.1.

Si l'arête est de bord, retourner l'arête tel que décrit à la section 3.1.2.

Si un des éléments créés a une aire presque nulle, passer à l'arête suivante.

Si la nouvelle arête créée a ses deux sommets sur le bord, passer à l'arête suivante.

Calculer la nouvelle qualité minimale des éléments de la coquille.

Si la qualité minimale est augmentée, faire :

Retourner l'arête dans le maillage global et passer à l'arête suivante.

Sinon, passer à l'arête suivante sauf si l'arête initiale avait ses deux sommets

sur le bord et que la nouvelle arête créée a au moins un de ses sommets interne.

Retourner l'arête dans le maillage global.

Passer à l'arête suivante.

Fin.

TAB. 3.2 – Algorithme de retournement d'arêtes.

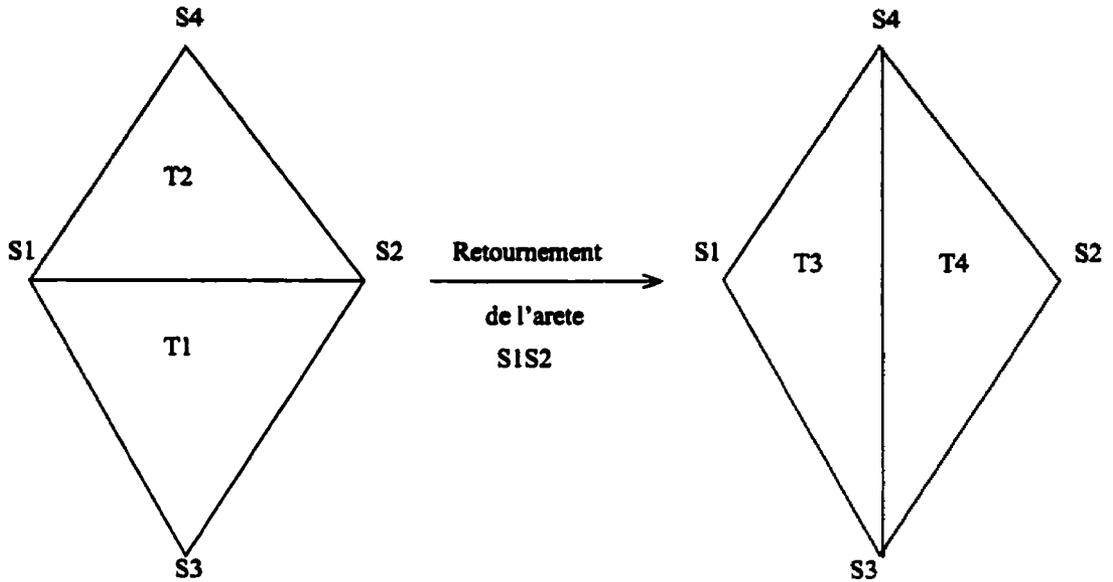


FIG. 3.1 – Retournement d'une arête interne.

Pour faire le retournement de cette arête, on détruit d'abord l'arête $\overline{S1S2}$ et donc les deux triangles T1 et T2. Ensuite, on construit la nouvelle arête $\overline{S3S4}$ et les deux nouveaux triangles T3 et T4.

3.1.2 Retournement d'une arête de bord

On dira qu'une arête est de bord si elle fait partie de la frontière Γ du maillage. Elle est donc adjacente à un seul élément triangulaire. Dans la figure 3.2, l'arête $\overline{S1S2}$ est de bord.

Pour faire le retournement de cette arête, on détruit d'abord le triangle T1. On crée ensuite un nouveau sommet S4 sur l'arête $\overline{S1S2}$ à égale distance de S1 et S2 dans la métrique (c'est le seul endroit où la métrique joue un rôle dans les opérations de retournement). On construit la nouvelle arête $\overline{S3S4}$ et les deux nouveaux triangles T2 et T3.

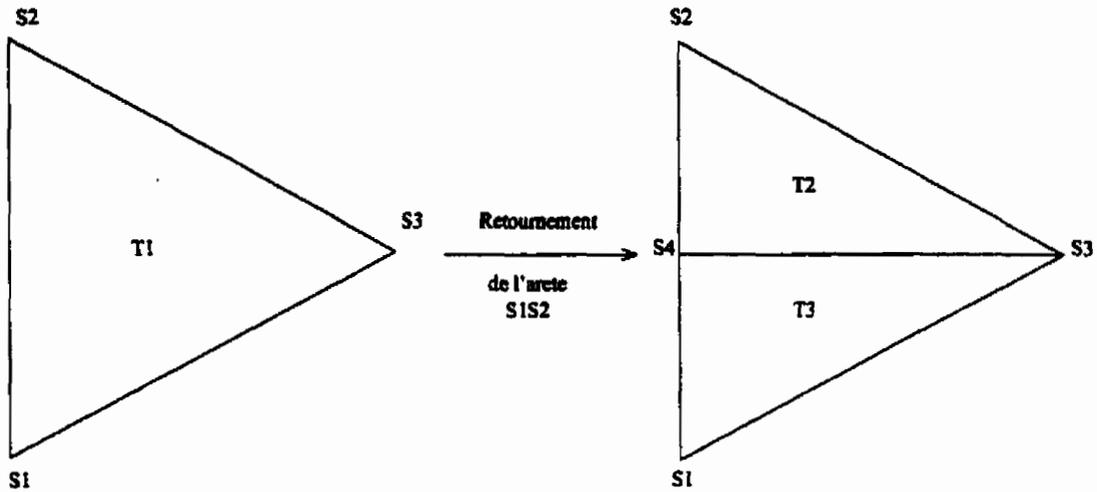


FIG. 3.2 – Retournement d'une arête de bord.

3.2 Déplacement de sommets

La seconde façon d'améliorer la qualité Q_K (2.12) des éléments consiste à déplacer les sommets du maillage afin d'en modifier la forme. Évidemment, cela a pour conséquence de changer les longueurs des arêtes dans la métrique de l'erreur. On pourrait ainsi utiliser cette opération pour se rapprocher du critère $h(\mathbf{x}) = Cte$ (Sec. 2.2.6). Nous avons donc choisi de tenir compte de ces deux critères pour obtenir un maillage plus adéquat. Un peu de notation nous aidera à préciser comment.

Pour un sommet s donné et pour la coquille correspondante, on notera :

- $\{a_i\}_{i=1}^k$ les arêtes connectées au sommet s .
- $l_{a_i}^e$ la longueur euclidienne de a_i .
- $l_{a_i}^m$ la longueur de a_i dans la métrique.
- La moyenne des longueurs euclidiennes : $\bar{l}^e = \frac{1}{k} \sum_{i=1}^k l_{a_i}^e$.
- La moyenne des longueurs dans la métrique : $\bar{l}^m = \frac{1}{k} \sum_{i=1}^k l_{a_i}^m$.
- \vec{a}_i l'arête a_i orientée du sommet s vers le sommet du bord de la coquille.
- $\xi_i = \frac{l_{a_i}^m}{l_{a_i}^e}$.
- $p = \sum_{i=1}^k \xi_i$.

Notre but est maintenant de calculer un vecteur déplacement $\vec{\Delta}$. Si nous cherchions à construire un maillage parfaitement isotrope, nous voudrions déplacer le sommet s vers le barycentre de sa coquille, mais ce n'est pas ce que nous recherchons. Nous allons plutôt considérer que chacun des sommets représente une masse qui agit sur la masse située en s . Les arêtes \vec{a}_i représentent maintenant des forces d'intensité variable, qui dépendent de la longueur euclidienne et de la métrique. Le vecteur $\vec{\Delta}$ devient un déplacement résultant que nous écrivons

$$\vec{\Delta} = \sum_{i=1}^k C_i^m C_i^e \vec{a}_i$$

où C_i^m et C_i^e sont des poids associés au déplacement dans la direction \vec{a}_i . Le poids C_i^m uniformise la longueur des arêtes dans la métrique tandis que C_i^e améliore la qualité des éléments en uniformisant la longueur euclidienne des arêtes.

On choisira $C_i^m = \frac{l_{a_i}^m - \bar{l}^m}{\bar{l}^m}$ qui forcera les longueurs $l_{a_i}^m$ à se rapprocher de \bar{l}^m et $C_i^e = \frac{\bar{l}^e}{l_{a_i}^e}$ qui aura un effet amplificateur ou amortisseur sur le déplacement lorsque $l_{a_i}^e$ sera loin de la moyenne \bar{l}^e .

Alors, on peut réécrire

$$\vec{\Delta} = \sum_{i=1}^k \vec{a}_i \cdot \frac{l_{a_i}^m - \bar{l}^m}{\bar{l}^m} \cdot \frac{\bar{l}^e}{l_{a_i}^e}$$

qui se simplifie pour donner

$$\vec{\Delta} = \frac{1}{p} \sum_{i=1}^k \vec{a}_i \cdot \left(\xi_i - \frac{\bar{l}^m}{l_{a_i}^e} \right). \quad (3.1)$$

Pour un exemple géométrique voir la figure 3.3.

Ainsi, chaque sommet subit un déplacement $\vec{\Delta}$. Par contre, il faut vérifier que le sommet ne sort pas de la coquille pour ne pas créer des éléments superposés ou des arêtes dont la longueur euclidienne ne respecte pas le critère $l \in [l_{min}^e, l_{max}^e]$.

Le déplacement de sommets n'a pas un effet très marqué sur le maillage, mais c'est la seule opération qui agit sur les sommets déjà existants sans en modifier le nombre. C'est

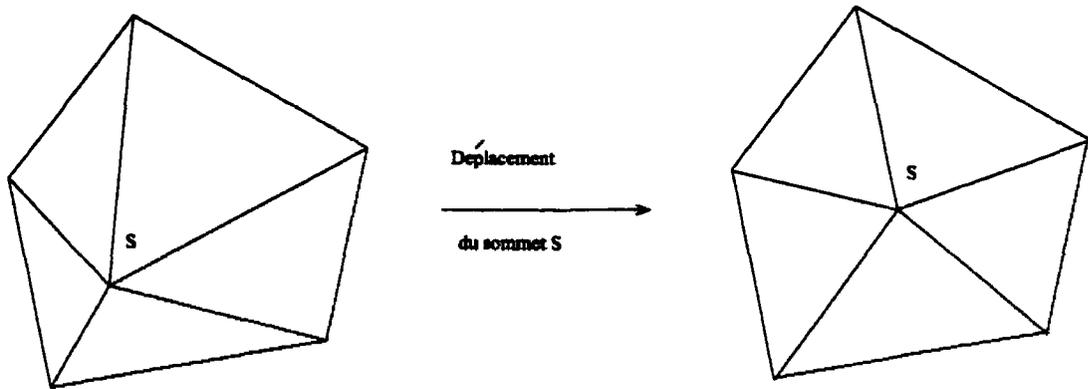


FIG. 3.3 – Déplacement d'un sommet.

une opération qui est très utile pour régulariser le maillage lorsqu'elle est combinée au retournement d'arêtes.

3.3 Raffinement d'arêtes

Cette opération, basée sur la philosophie LIBOM, consiste à subdiviser les arêtes qui portent une erreur trop grande afin de la redistribuer sur plusieurs arêtes.

En pratique, la mesure de l'erreur est donnée par la longueur de l'arête dans la métrique (voir section 2.2.5) et la diminution de cette erreur découle de la subdivision des arêtes. Pour une arête donnée, les deux arêtes créées par cette subdivision porteront donc la moitié de l'erreur de l'arête originale et seront ainsi plus petites dans la métrique.

Le raffinement d'arêtes est une opération coûteuse, car elle crée beaucoup de nouveaux sommets, de nouvelles arêtes et de nouveaux éléments. On doit bien contrôler le raffinement d'arêtes afin d'éviter que trop d'éléments soient ajoutés au maillage ce qui affecterait grandement le coût de la résolution. Si cette opération est bien effectuée, seules les zones où l'erreur est élevée seront raffinées. En d'autres mots, la concentration des éléments sera grande là où la métrique sera la plus élevée.

$\|\vec{\Delta}\|_{max} = 1.5$, le déplacement maximal permis.
 $NbIter_{max} = 10$, le nombre maximum d'itérations.
 Pour chaque sommet s du maillage global, faire :
 ConfigurationValide = FAUX.
 $x^* = s$, le déplacement optimal.
 $\Omega = 1.0$.
 $NbIter = 0$, le nombre d'itérations exécutées.
 Si le sommet est un coin (0D) ou de bord (1D) passer au sommet suivant.
 Construire une coquille autour du sommet s .
 Pour les arêtes $\{a_i\}_{i=1}^k$ connectées au sommet s , calculer :

- $l_{a_i}^e, l_{a_i}^m, \bar{l}^e$ et \bar{l}^m tel que décrit à la section 3.2.
- la qualité minimale des éléments de la coquille avec la formule (2.12).
- \vec{a}_i, ξ_i, p et le déplacement $\vec{\Delta} = \frac{1}{p} \sum_{i=1}^k \vec{a}_i \cdot (\xi_i - \frac{\bar{l}^m}{l_{a_i}^e})$.

 Si $\Omega \cdot \|\vec{\Delta}\| > \bar{l}^e \cdot \vec{\Delta}_{max}$, poser $\Omega = \frac{\bar{l}^e \cdot \vec{\Delta}_{max}}{\|\vec{\Delta}\|}$.
 Tant que $NbIter < NbIter_{max}$ et ConfigurationValide est FAUX, faire :
 ConfigurationValide = VRAI.
 $x^* = s + \vec{\Delta} \cdot \Omega$.
 Si x^* sort de la coquille, ConfigurationValide = FAUX.
 Si la qualité minimale des éléments est diminuée, ConfigurationValide = FAUX.
 S'il y a création d'éléments superposés, ConfigurationValide = FAUX.
 Si ConfigurationValide = FAUX, poser $\Omega = \frac{\Omega}{2}$ et $NbIter = NbIter + 1$.
 Fin.
 Si $\frac{\|x^* - s\|}{\bar{l}^e} < \epsilon$ i.e. le déplacement est négligeable, passer au sommet suivant.
 Si une des nouvelles arêtes créées a une longueur euclidienne $< l_{min}^e$ ou $> l_{max}^e$
 (critère (2.11)), passer au sommet suivant.
 Poser $s = x^*$ et passer au sommet suivant.
 Fin.

TAB. 3.3 – Algorithme de déplacement de sommets.

Dans ce contexte, le maillage obtenu pourra avoir des triangles étirés ou non selon la définition de la métrique. Ce qui donne un contrôle complet sur l'anisotropie, par choix de la métrique. Pour une métrique d'erreur, il est très probable que certaines régions du maillage soient fortement anisotropes tandis que d'autres soient fortement isotropes.

Le raffinement est contrôlé par la fonction $h(\boldsymbol{x})$ définie à la section 2.2.6. Si la longueur de l'arête a dans la métrique l_a^m ne satisfait pas l'inégalité (2.10), ou plus précisément, si $l_a^m > l_c^m \cdot \delta_l$, on effectue le raffinement. Ici, nous avons utilisé $\delta_l = 1.5$. Par ailleurs, et bien que nous acceptions l'anisotropie, le critère de longueur euclidienne minimale admissible sera tout de même appliqué. Il servira entre autre à stopper le raffinement excessif dans les zones où l'erreur est infinie i.e. aux abords d'une singularité. Cela évitera d'avoir des arêtes trop petites qui introduiraient des instabilités numériques dans les calculs.

Dans cette section, nous détaillerons l'algorithme de raffinement d'arêtes au tableau 3.4 alors que les deux cas typiques de raffinement d'une arête (intérieure et sur le bord du domaine) seront détaillés aux sections 3.3.1 et 3.3.2 respectivement. Les figures 3.4 et 3.5 illustrent les deux cas possibles.

3.3.1 Raffinement d'une arête interne

Pour faire le raffinement d'une arête interne, on crée un nouveau sommet $S5$ qui est en fait le milieu de l'arête $\overline{S1S2}$ dans la métrique donnée. On détruit ensuite les triangles $T1$ et $T2$. On construit par la suite les 4 arêtes ($\overline{S1S5}$, $\overline{S2S5}$, $\overline{S3S5}$ et $\overline{S4S5}$) et les 4 triangles ($T3$, $T4$, $T5$ et $T6$). On peut se rapporter à la figure 3.4.

3.3.2 Raffinement d'une arête de bord

Le raffinement de bord est illustré à la figure 3.5 et se détaille comme suit. On crée un nouveau sommet $S4$ situé au milieu de l'arête $\overline{S1S2}$ dans la métrique M . On détruit

Pour chaque arête a du maillage global (incluant les nouvelles arêtes créées), faire :

Calculer la longueur l_a^m de a dans la métrique.

Si $l_a^m \leq l_c^m \cdot \delta_l$ (critère (2.10)), passer à l'arête suivante.

Trouver le point milieu m de l'arête $a = \overline{a_1 a_2}$ dans la métrique.

Calculer les longueurs euclidiennes des arêtes $\overline{a_1 m}$ et $\overline{m a_2}$,
notées respectivement $l_{a_1 m}^e$ et $l_{m a_2}^e$.

Si $l_{a_1 m}^e < l_{min}^e$ ou $l_{m a_2}^e < l_{min}^e$ (critère (2.11)), passer à l'arête suivante.

Si l'arête a est interne, raffiner l'arête tel que décrit à la section 3.3.1.

Si l'arête a est de bord, raffiner l'arête tel que décrit à la section 3.3.2.

Réintégrer les changements dans le maillage global.

Passer à l'arête suivante.

Fin.

TAB. 3.4 – Algorithme de raffinement d'arêtes.

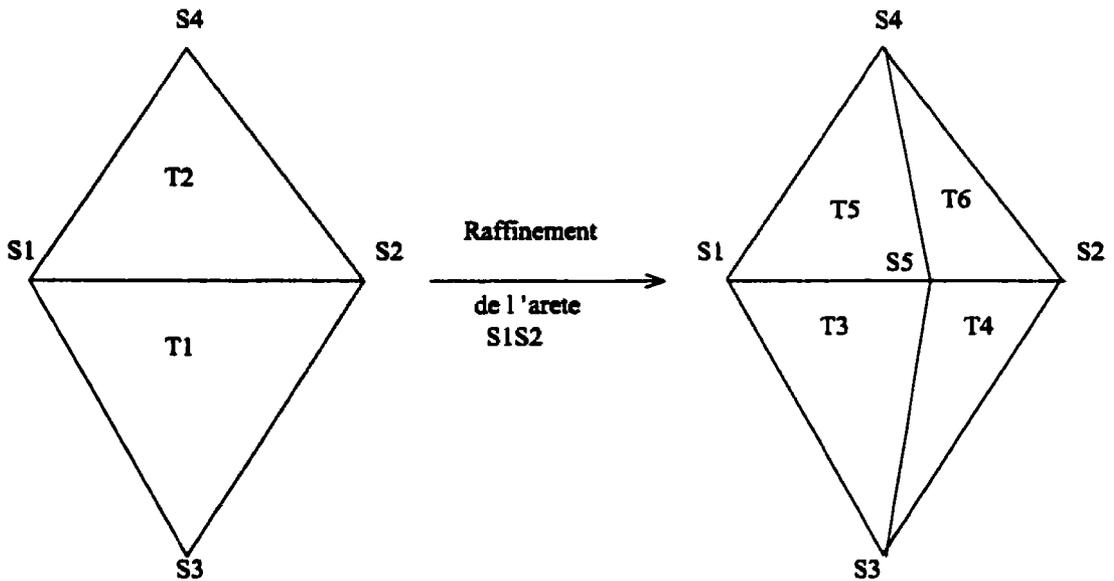


FIG. 3.4 – Raffinement d'une arête interne.

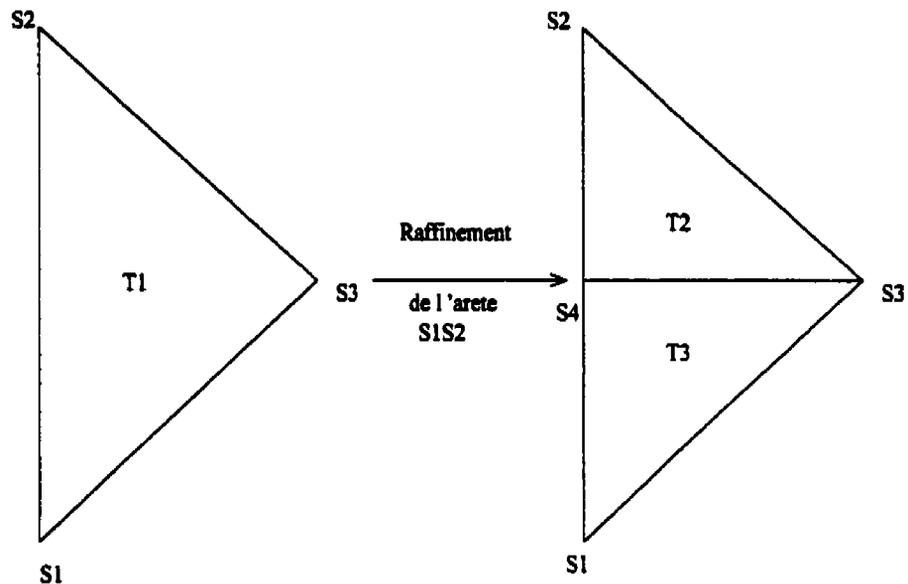


FIG. 3.5 – Raffinement d'une arête de bord.

ensuite le triangle $T1$. On construit par la suite l'arête $\overline{S3S4}$ et les 2 triangles ($T2$ et $T3$).

3.4 Déraffinement d'arêtes

Le déraffinement d'arêtes est basé sur la philosophie LIBOM et consiste à enlever les arêtes qui, mesurées dans la métrique, sont trop courtes. Par mesure de simplicité, nous allons utiliser un **déraffinement de sommets** pour effectuer cette opération. En effet, le déraffinement d'arêtes est plus facile à implémenter en utilisant un déraffinement de sommets, i.e. en enlevant les sommets qui délimitent une arête porteuse d'une erreur trop petite.

Cette approche demande beaucoup de soin si l'on veut éviter la création d'éléments superposés. Nous proposerons un algorithme ainsi que quelques arguments heuristiques pour justifier sa convergence.

Comme le raffinement, le déraffinement est contrôlé par la fonction $h(\mathbf{x})$ définie à la section 2.2.6. Si la longueur de l'arête dans la métrique de l'erreur l_a^m ne satisfait pas

l'inégalité (2.10), ou plus précisément si $l_a^m < \frac{l_a^m}{\delta_l}$, on effectue le déraffinement (nous utilisons $\delta_l = 1.5$). Par contre, on ne déraffine pas si $l_a^e > l_{max}^e$ où l_{max}^e est la longueur euclidienne maximale tolérée. De plus, on ne permettra pas le déraffinement des sommets de coin (sommets 0D) pour des raisons évidentes. Par contre, on pourra déraffiner les sommets de bord (sommets 1D). Ce choix sera un paramètre de l'algorithme. À noter qu'aucune restriction n'est mise sur les sommets intérieurs (sommets 2D).

Le déraffinement d'arêtes est une opération coûteuse, car elle exige beaucoup de calculs durant la reconstruction de la coquille et qu'elle crée de nouveaux éléments et de nouvelles arêtes. On doit bien contrôler le déraffinement d'arêtes, car trop d'éléments et de sommets peuvent être enlevés. Le fait d'enlever trop d'éléments peut conduire à une solution très grossière. De plus, la diminution du nombre de sommets contribue à diminuer la qualité de l'estimateur car la métrique de l'erreur est définie en chaque sommet. Cette perte d'information doit être contrôlée et c'est pourquoi on doit effectuer le raffinement avant le déraffinement. En outre, le choix d'une petite valeur du paramètre l_{max}^e peut permettre de diminuer cet effet.

Dans cette section, nous détaillerons l'algorithme de déraffinement d'arêtes au tableau 3.5. Le sous-algorithme de remaillage de la coquille vide sera décrit au tableau 3.6 et sa validité sera étudiée à la section 3.4.1. La figure 3.6 illustre un exemple de déraffinement.

Pour chaque arête a du maillage global, faire :

Si la longueur l_a^m de $a = \overline{a_1 a_2}$ dans la métrique, satisfait $l_a^m < \frac{l_a^e}{\delta_l}$, faire :

Si a_1 est un sommet interne, marquer a_1 .

Sinon

Si a_2 est un sommet interne, marquer a_2 .

Sinon

Si a_1 est un sommet de bord, mais pas de coin et que

DeraffineSommet1D est VRAI, marquer a_1 .

Si a_2 est un sommet de bord, mais pas de coin et que

DeraffineSommet1D est VRAI, marquer a_2 .

Fin.

Pour chaque sommet marqué \bar{s} du maillage global, faire :

Construire une coquille autour du sommet \bar{s} .

Pour chaque arête a de la coquille, calculer la longueur l_a^m de a dans la métrique et la longueur euclidienne l_a^e .

Calculer la longueur euclidienne maximale $\bar{s}_{l_{max}^e}$ des arêtes de la coquille.

Si $\bar{s}_{l_{max}^e} > l_{max}^e$ (critère (2.11)), passer au sommet suivant.

Si une des arêtes a de la coquille satisfait $l_a^m < \frac{l_a^e}{\delta_l}$, enlever \bar{s} de la coquille et reconstruire cette coquille vide selon l'algorithme du tableau 3.6.

Sinon, passer au sommet suivant.

Fin.

TAB. 3.5 – Algorithme de déraffinement de sommets.

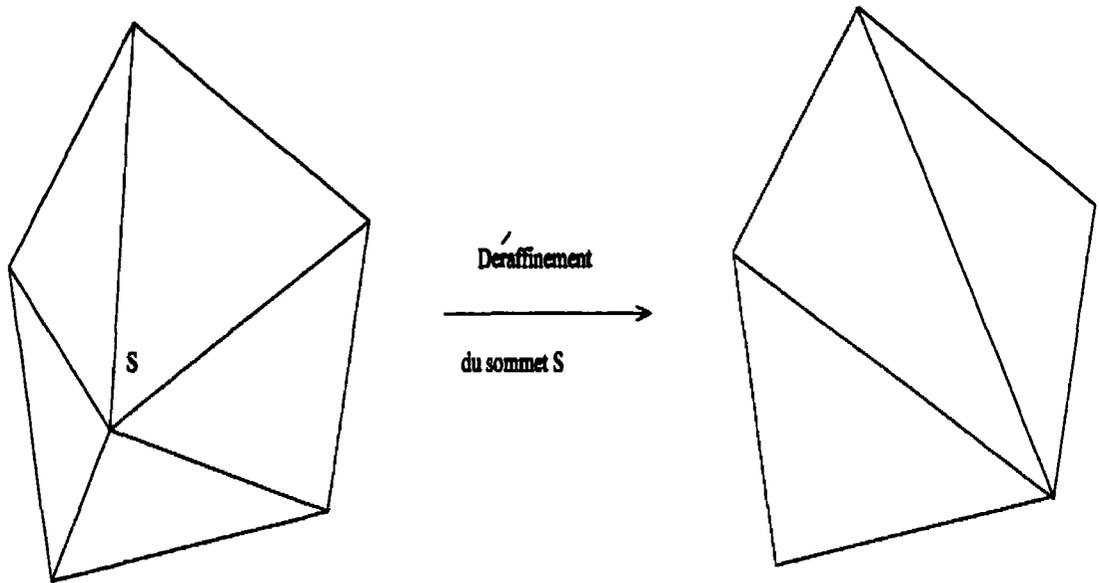


FIG. 3.6 – Déraffinement d'un sommet.

3.4.1 Algorithme de reconstruction d'une coquille vide

Lemme 1 *Pour chaque courbe polygonale simple et fermée Γ à n sommets, ($n \geq 4$), il existe une paire s_1, s_2 de sommets non consécutifs pour lesquels l'arête $\overline{s_1 s_2}$ divise Γ en deux courbes polygonales simples et fermées ayant au plus $n - 1$ sommets.*

Démonstration : Remarquons d'abord que, pour deux sommets quelconques $\{s_i, s_j\} \in \Gamma$, ou bien l'arête $\overline{s_i s_j}$ a la propriété désirée ou bien elle coupe Γ en au moins un autre point qui est soit un sommet, soit un point intérieur d'une autre arête.

Puisque la somme des angles intérieurs d'un polygone à n sommets est égale à $(n - 2)\pi$, il existe nécessairement un sommet s pour lequel l'angle intérieur est inférieur à π . Notons alors s_p le sommet qui précède s et s_r celui qui le suit sur Γ (l'ordre de parcours n'a pas d'importance). Ou bien l'arête $\overline{s_p s_r}$ est celle qu'on cherche, ou bien elle coupe Γ au moins une fois et donc au moins un sommet de Γ se trouve dans l'intérieur du triangle $\Delta_{s_p s s_r}$.

Définissons la distance d'un point p au sommet s par $D_s(p) = \|\pi_d p - s\|_2$ où $\pi_d p$ est

On crée :

$$\mathcal{L} = \{l_m\}_{m=1}^n.$$

$i = 1.$

Tant que $i \leq n - 3$, faire :

On met à jour $j = 1.$

Tant que $j \leq n - i + 1$, faire :

Si l'angle interne en l_j satisfait $\angle l_{j-1}l_jl_{j+1} < \pi$ et que la fermeture du triangle $\Delta l_{j-1}l_jl_{j+1}$ ne contient aucun sommet de $\mathcal{L} \setminus \{l_{j-1}, l_j, l_{j+1}\}$, faire :

- a) On construit l'arête $\overline{l_{j-1}l_{j+1}}$ du triangle $\Delta l_{j-1}l_jl_{j+1}$.
- b) On construit le triangle $\Delta l_{j-1}l_jl_{j+1}$.
- c) On enlève le sommet l_j dans \mathcal{L} .
- d) On renumérote \mathcal{L} de 1 à $n - i$.
- e) On met à jour $i = i + 1$.
- f) On met à jour $j = n - i + 3$.

Sinon faire :

- a) On met à jour $j = j + 1$.

Fin.

Fin.

TAB. 3.6 - Algorithme de triangulation d'une courbe Γ polygonale simple et fermée.

la projection de p sur la droite $d = s\vec{p} + s\vec{s}$. Cherchons $\mathcal{X} = \operatorname{argmin}_{y \in \Gamma} D_s(y)$ l'ensemble des points de Γ les plus près de s où Γ est l'intersection de Γ et l'intérieur du triangle $\Delta s_p s s_s$. La courbe Γ étant compacte et la fonction à minimiser continue, $\mathcal{X} \neq \emptyset$. \mathcal{X} comprend donc des sommets et/ou des arêtes complètes de Γ . Dans ce second cas, \mathcal{X} contient nécessairement les sommets qui délimitent ces arêtes. Donc, nous pouvons choisir un sommet $x \in \mathcal{X}$. Avec ce choix, l'arête \overline{sx} divise l'intérieur de Γ en deux sous-domaines car \overline{sx} est à l'intérieur de Γ et ne coupe aucune arête ni sommet. C'est l'arête cherchée. ■

Corollaire 1 *Soit Γ une courbe polygonale simple et fermée à n sommets, où $n \geq 4$. Alors il existe maillage valide de Γ composé de triangles.*

Démonstration : Appliquer le lemme 1 de façon répétée sur les sous-domaines créés jusqu'à ce que le dernier n'ait plus que trois sommets. ■

Théorème 1 *Soit Γ une courbe polygonale simple et fermée de sommets $\mathcal{L} = \{l_m\}_{m=1}^n$ où $n \geq 3$ et où l'on suppose que l'ordre correspond au parcours de Γ dans le sens antihoraire. En outre, on convient de prolonger cette liste en une liste infinie pour laquelle le sommet $l_k \in \mathcal{L}$ d'indice $k \in \mathbb{N}$ coïncide avec le sommet $l_{(k \bmod n) + 1} \in \mathcal{L}$. Alors, l'algorithme du tableau 3.6 conduit toujours à une triangulation valide du domaine borné délimité par Γ .*

Démonstration : Notons d'abord que, sous les hypothèses géométriques satisfaites par Γ , il est toujours possible d'ordonner les sommets tels que désiré. Montrons que l'algorithme du tableau 3.6 converge toujours vers une triangulation. Pour ce faire, nous allons démontrer par récurrence que la proposition : "Si $n \geq 3$, il existe un sommet $l_i \in \mathcal{L}$ tel que l'angle interne associé $\angle l_{i-1}l_i l_{i+1}$ satisfait $\angle l_{i-1}l_i l_{i+1} < \pi$ et qu'aucun sommet de $\mathcal{L} \setminus \{l_{i-1}, l_i, l_{i+1}\}$ ne se trouve dans la fermeture du triangle $\Delta l_{i-1}l_i l_{i+1}$ " est toujours vraie. Si c'est le cas, nous obtiendrons un premier triangle que nous pourrons éliminer pour ensuite appliquer le même raisonnement au domaine complémentaire. Il est important de noter que cette façon de procéder nous assure que nous ne créerons pas de triangles superposés car aucun sommet ne sera dans $\overline{\Delta l_{i-1}l_i l_{i+1}}$, et aucune arête ne le coupera.

Pour $n = 3$, la proposition est triviale. Supposons qu'elle est vraie pour $n > 3$ et montrons qu'elle l'est pour $n+1$. Selon le lemme 1, il existe une paire de sommets $\{s_i, s_j\} \in \Gamma$ tel que $\overline{s_i s_j}$ divise l'intérieur de Γ en deux sous-domaines délimités par deux courbes de Jordan polygonales simples et fermées Γ_1 et Γ_2 ayant chacune au plus n sommets.

Si Γ_1 ou Γ_2 n'a que trois sommets, ils constituent le triplet cherché. Sinon, considérons, sans perte de généralité, le cas de Γ_1 . Puisque cette courbe n'a que $3 < p \leq n$ sommets, par hypothèse de récurrence, elle doit avoir un sommet $l_k \in \mathcal{L}$ pour lequel $\angle l_{k-1}, l_k, l_{k+1} < \pi$ et l'intérieur de $\Delta l_{k-1} l_k l_{k+1}$ ne contient aucun sommet ni arête de Γ . Si les sommets $\{l_{k-1}, l_k, l_{k+1}\}$ sont consécutifs sur la courbe Γ de départ, on a, de nouveau, trouvé le triplet cherché. Dans le cas contraire, il faut que $l_k = s_i$ ou $l_k = s_j$, disons s_i . On utilise alors l'arête $\overline{l_{k-1} l_{k+1}}$ pour déterminer une nouvelle courbe Γ_3 qui ne contient pas s_j et donc n'a pas plus de $n - 1$ sommets.

On recommence alors le raisonnement précédent jusqu'à ce que le sommet intermédiaire l_k satisfasse $l_k \notin \{s_i, s_j\}$ ou bien que $n = 3$. Dans les deux cas, on aura trouvé un triplet de sommets consécutifs sur Γ qui satisfait l'hypothèse de récurrence. ■

3.5 Séquence d'adaptation

À la lumière des sections précédentes, nous pouvons détailler l'étape ADAPTATION de la séquence globale définie au tableau 3.1, étape qui combine les quatre opérations décrites précédemment. Comme nous en avons discuté à la section 3.4, le déraffinement d'arêtes est toujours effectué après le raffinement. Ceci permet de conserver plus de sommets qui servent de support à la métrique et de ne pas dégrader inutilement celle-ci. On décrira plus en détails les choix qui nous ont amené à choisir cette séquence dans la section 4.4. On définira une séquence de nettoyage comme une combinaison des opérations de déplacements de sommets et de retournement d'arêtes. Cette séquence de nettoyage a pour effet de rendre le maillage courant plus régulier. On débute donc en raffinant jusqu'à la

tolérance voulue, puis, on fait trois séquences de nettoyage afin de régulariser le maillage raffiné. Par la suite, on déraffine pour diminuer le nombre de ddls et on termine en régularisant de nouveau le maillage. Cette séquence sera utilisée pour résoudre les problèmes d'éléments finis du chapitre 5.

Si on définit la séquence de nettoyage comme suit :

$$\text{séquence de nettoyage} \left\{ \begin{array}{l} \text{a. retournement d'arêtes,} \\ \text{b. déplacement de sommet,} \\ \text{c. retournement d'arêtes,} \\ \text{d. déplacement de sommet,} \end{array} \right.$$

alors, la séquence d'adaptation peut être décrite ainsi :

$$\text{Séquence d'adaptation} \left\{ \begin{array}{l} 1. \text{ raffinement d'arêtes} \\ 2. \text{ séquence de nettoyage} \\ 3. \text{ déraffinement d'arêtes} \\ 4. \text{ séquence de nettoyage} \\ 5. \text{ séquence de nettoyage} \\ 6. \text{ séquence de nettoyage} \end{array} \right. \quad (3.2)$$

Chapitre 4

Implémentation d'un code d'adaptation en C++

4.1 Le GIREF et MEF++

Le développement du logiciel requis par ce travail a été fait dans l'environnement MEF++, c'est-à-dire l'environnement logiciel de recherche du GIREF (Groupe Interdisciplinaire de Recherche en Éléments Finis). Le GIREF est un centre de recherche universitaire qui regroupe des chercheurs travaillant dans le domaine de la modélisation et des méthodes numériques, plus particulièrement sur la méthode des éléments finis. Tous ces chercheurs ont en commun un même outil de simulation numérique : la méthode des éléments finis. Depuis 1995, le groupe développe un logiciel commun d'éléments finis nommé MEF++ ("Méthode d'Éléments Finis en C++").

Ce logiciel est avant tout un outil de calcul très complexe qui permet de résoudre des EDP par la MEF. Il est assez général et flexible pour que les mathématiciens, les ingénieurs et les physiciens puissent l'utiliser pour leurs simulations numériques. Cette qualité a été assurée par la création de bibliothèques communes dont les différents modules ont pour fin la gestion des systèmes linéaires de grandes tailles, leurs résolution, la construction

d'espaces d'éléments finis, celles des maillages ainsi que la manipulation des champs de calculs. De plus, une gestion suffisamment abstraite des conditions aux limites, la possibilité de manipuler des champs analytiques ainsi que de varier les formulations variationnelles permettent de satisfaire aussi bien les besoins de la mécanique des fluides que ceux de la mécanique des solides. L'adaptation fait aussi partie intégrante de MEF++ et c'est ce module qui a été notre point de départ pour l'implémentation des opérations décrites au chapitre 3. L'utilisation des bibliothèques existantes a facilité l'implémentation et la validation des algorithmes d'adaptation. Il faut cependant noter que ce logiciel n'inclut pas les interfaces graphiques pour initialiser les données et visualiser les solutions. De tels logiciels sont découplés de MEF++ pour que ce dernier reste un calculateur. Par exemple, la visualisation des résultats et la création des images ont été faites avec le logiciel VU du CERCA (Centre de Recherche en Calcul Appliqué) de l'université de Montréal qui est un outil convivial de visualisation bi- et tridimensionnel des solutions d'un calcul par éléments finis.

MEF++ utilise le plein potentiel du langage C++ en maximisant l'utilisation de la méthodologie orientée objet. Cette approche favorise l'encapsulation des données et des fonctions associées dans un artifice nommé **classe**. Notons qu'un objet est une réalisation concrète d'une classe. L'accès aux variables, appelées membres, se fait via les fonctions membres associées. Ainsi, les données sont protégées et la classe forme un ensemble complet en elle-même. On n'a donc pas besoin de connaître le fonctionnement interne d'une classe pour l'utiliser. Cela favorise la réutilisation et minimise la duplication de code et, par le fait même, les erreurs potentielles. C'est grâce à ces avantages que la majeure partie du travail des autres programmeurs et chercheurs peut être réutilisé. Cette possibilité de réutilisation sera exploitée avec succès et on la décrira plus en détails dans la section 4.2. L'exemple qui suit illustre bien la réutilisabilité de MEF++ et l'encapsulation des méthodes et des données. Cet exemple est tiré de la classe `FormStokesNum2D` qui est utilisée dans le programme `StokesNum2D` (Sect. 4.3.2). On concrétise cette classe par un objet nommé `FormulationStokes` et on utilise les fonctions membres pour fixer les valeurs

des paramètres comme la viscosité et pour définir les champs de calculs :

```
FormulationStokes.asgnViscositeConstante(Viscosite) ;
```

```
FormulationStokes.asgnChamps(Vitesse,Pression,ForceVolumique,Geometrie) ;
```

Ceci fait, la construction de la matrice et du vecteur de droite ne requiert qu'un simple appel d'une fonction membre :

```
FormulationStokes.assembleSysteme(MatriceGlobale, VectDroite) ;
```

Le logiciel MEF++ est très complexe et, pour en avoir une vue d'ensemble adéquate et en connaître le fonctionnement global, on doit y consacrer plusieurs heures. L'implication de l'auteur dans la mise au point de modules dans MEF++ remonte bien avant ce travail. En effet, il a contribué à la création du module de traitement des conditions limites, à la validation des calculs numériques et à la mise au point de certaines interfaces entre MEF++ et d'autres logiciels d'éléments finis. Malgré cette connaissance de première main, il est bon de noter que le module d'adaptation était, pour lui, dans une région inexplorée du code.

4.2 Réutilisation des bibliothèques de MEF++

La majeure partie des bibliothèques de MEF++ qui ont été utilisées, sont en fait des bibliothèques de base et n'ont pas été modifiées. Par contre, beaucoup d'efforts ont été mis pour comprendre leur fonctionnement. Voici une courte liste commentée de quelques-unes de ces bibliothèques :

1. Adaptation : bibliothèque qui contient toutes les classes propres à l'adaptation de maillages.
2. Champs : bibliothèque volumineuse qui contient les définitions des champs définis sur un maillage, comme par exemple, ChampVectoriel3D, ChampTensorielO2Sym, Champ-

Scalaire, etc.

3. **Coquilles** : librairie qui permet la création et la manipulation de sous-maillages. Cette dernière est très utilisée pour l'adaptation.
4. **ConditionsLimites** : librairie qui sert à initialiser et à assigner les conditions aux limites.
5. **Elements** : cette librairie contient les définitions et les constructions relatives aux espaces d'éléments finis.
6. **Formulation** : librairie qui contient les déclarations des termes de formulation pour des problèmes de mécanique des fluides et des solides. On utilisera les termes `StokesNum2D` et `MaxwellNum2D`.
7. **Maillage** : cette librairie contient tout ce qui a trait aux maillages.
8. **Num** : cette librairie contient divers outils et classes facilitant les calculs et les manipulations numériques, e.g. `Vecteur2D`, `TenseurO2Sym`, `produitMatriceVecteur`.
9. **Petsc** : interface C++ avec la librairie PETSc (Portable Extensible Toolkit for Scientific Computation), externe à MEF++, qui permet de manipuler et résoudre des systèmes linéaires de grande taille.

4.3 Développement en C++

4.3.1 Modification et création de bibliothèques

Pour implanter les algorithmes d'adaptation, les principales classes que nous avons modifiées sont les classes d'estimation d'erreur et les classes concernant l'utilisation de la métrique, plus particulièrement le calcul de la qualité des éléments. De plus, nous avons dû ajouter, dans la librairie `Num`, diverses fonctions servant notamment dans les algorithmes d'adaptation telles que celle servant au calcul de l'orientation des triangles et celle servant à la recherche du sommet opposé à une arête.

La classe d'adaptation de maillages en dimension 2 (`AdaptationMaillage2D`) est la classe principale que nous avons dû créer et enrichir. Au départ, nous nous sommes basés sur la classe `AdaptationMaillage3D` qui a pour fonction d'adapter un maillage formé d'éléments tridimensionnels comme les tétraèdres. Nous avons d'abord transposé en dimension 2 certains algorithmes de la classe. Cette étape fut très importante car elle nous a permis de comprendre les concepts et le fonctionnement sous-jacent à l'adaptation de maillages et à la manipulation des maillages et des métriques. Mais, nous nous sommes rapidement aperçus que ces algorithmes étaient plus ou moins performants en dimension 2 (e.g. le raffinement d'arêtes) et que le déraffinement de sommets ne pouvait pas être transposé intégralement.

Dans les paragraphes qui suivent, nous allons commenter les développements accomplis pour chacune des quatre opérations de base décrites au chapitre 3.

Le déraffinement de sommets (Sect. 3.4) fut propice à plusieurs essais d'algorithmes. Tout d'abord, nous avons décidé d'utiliser les idées de la classe `AdaptationMaillage3D`. Or, dans certains cas, des éléments superposés étaient créés lors du remaillage de la coquille. Pour remédier à ce problème, une étape supplémentaire devait être ajoutée pour corriger ces éléments superposés et cette idée ne nous plaisait pas. Nous avons donc opté pour un algorithme de remaillage qui ne crée pas d'éléments superposés. Tout d'abord, l'algorithme fut construit de façon récursive pour explorer tout l'arbre des possibilités de remaillage et ainsi choisir le maillage avec la meilleure qualité minimale des éléments. Après plusieurs tests de performance, nous avons remarqué que cette opération était la plus coûteuse en temps de calculs pour l'adaptation et ce, par un facteur assez grand. Nous avons donc décidé de créer un algorithme qui trouve une solution de remaillage et ce, sans explorer tout l'arbre. En fait, cet algorithme explore un sous-ensemble particulier dans lequel il existe toujours une solution (Voir la section 3.4.1). Une autre raison qui justifiait ce changement était que le remaillage optimal serait de toute manière détruit par les opérations subséquentes, notamment le déplacement de sommets et le retournement

d'arêtes.

Le déplacement de sommets (Sect. 3.2) était facilement réutilisable en 2D et n'exigeait pas beaucoup de modifications. Le raffinement d'arêtes (Sect. 3.3) a été nettoyé et une optimisation a été faite par des vérifications peu coûteuses. Ces dernières déterminent si le raffinement est nécessaire pour une arête donnée. Le retournement d'arêtes (Sect. 3.1) fut, lui aussi, optimisé par l'ajout de vérifications mais nécessiterait encore quelques améliorations. De plus, des protections ont été ajoutées pour ne pas avoir d'arêtes dont les deux sommets sont sur le bord.

Des changements ont été apportés aux classes qui définissent la métrique, dans le but d'utiliser la définition (2.12) de la qualité Q_K des éléments.

La classe d'estimation d'erreur a connu quelques ajouts de nouveaux paramètres qui concernent l'adaptation. Parmi ceux-ci on compte notamment l_{min}^e et l_{max}^e . On trouve à la section 4.3.3, de plus amples détails à ce sujet.

4.3.2 Modification et création de programmes

Pour résoudre les problèmes concernant les fluides visco-élastiques qui sont détaillés au chapitre 5, les programmes `StokesNum2D` et `MaxwellNum2D` ont été utilisés et modifiés.

En premier lieu, la résolution du problème de Stokes (Sect. 5.4.1) est faite par le programme `testStokesNum2D` (un élément de la première étape dans la séquence globale du tableau 3.1) qui utilise le terme de formulation `StokesNum2D`. Ce programme a dû être modifié pour permettre la lecture des fichiers de paramètres et rediriger les résultats dans un fichier. Ces résultats, plus spécifiquement la solution en vitesse et les sorties graphiques, sont réutilisés par d'autres programmes.

Par la suite, le programme `testMaxwellNum2D` (un autre élément de la première étape dans la séquence globale du tableau 3.1) est utilisé avec les données de `testStokesNum2D` pour résoudre l'équation de comportement de Maxwell (Sect. 5.4.4). Plusieurs modifications ont été nécessaires pour relire ces données et le fichier de configuration de `testMaxwellNum2D`. De plus, les sorties ont été sauvegardées dans un format particulier, notamment le champ tensoriel de σ , le champs linéaire $\|\sigma\|$ et le fichier graphique.

Aussi plusieurs programmes ont été créés pour analyser les maillages et les solutions, en particulier pour en calculer l'erreur commise selon différentes normes.

Enfin, le programme d'adaptation de maillage nommé `AdaptationMaillage2DChampScalaireUsager` a dû être créé de toutes pièces. Il a pour but de trouver le meilleur maillage adapté en se basant sur le champs scalaire définissant l'erreur et ce, en effectuant la séquence d'adaptation (3.2). De plus, le fichier de paramètres "`testAdaptation2DMetriqueChampScalaireUsager.txt`" contient toutes les variables qui contrôlent le fonctionnement de l'adaptation. Par exemple, celui-ci détermine le choix entre lire les résultats de `testMaxwellNum2D` pour créer la métrique ou bien utiliser une métrique construite à partir d'un champs analytique pré-défini. Une description plus détaillée de ce fichier se trouve dans la section 4.3.3.

4.3.3 Fichier de paramètres pour l'adaptation

Le fichier contenant les paramètres relatifs au programme `testAdaptation2DMetriqueChampScalaireUsager` se nomme `testAdaptation2DMetriqueChampScalaireUsager.txt`. Il contient 22 lignes dont la moitié débutent par un "#" et contiennent des commentaires concernant la ligne subséquente. Un exemple d'un tel fichier est donné au tableau 4.1. Ce tableau est suivi d'une description plus détaillée des données fournies aux lignes de rang pair.

Ligne	Contenu du fichier testAdaptation2DMetriqueChampScalaireUsager.txt
L1	#Utiliser un champ analytique pour la metrique (oui=1 ou non=0)
L2	1
L3	#Expression analytique, f(x,y,z) de la metrique
L4	1/((x ² +y ²) ^(1/8) +0.08)
L5	#Fichier contenant un champ lineaire
L6	coucheLimiteMaxwellNormeSigma
L7	#Nombre d'adaptations (Entier > 0)
L8	1
L9	#Deraffine (oui=1 ou non=0)
L10	1
L11	#Deraffine sommet 1D (oui=1 ou non=0)
L12	0
L13	#Longueur desiree des aretes dans la metrique (Reel > 0)
L14	0.01
L15	#L'usager fournit la longueur courante a utiliser? (oui=1 ou non=0)
L16	0
L17	#Nombre de divisions de l'arete la plus longue (Reel > 1)
L18	4.0
L19	#Longueur minimale euclidienne (Reel > 0)
L20	0.005
L21	#Longueur maximale euclidienne (Reel > 0)
L22	0.3

TAB. 4.1 – Contenu du fichier testAdaptation2DMetriqueChampScalaireUsager.txt.

- L2) Paramètre d'initialisation de $h(\mathbf{x})$ (section 2.2.3). S'il vaut 1, on utilise l'expression algébrique de la ligne L4, sinon on importe les données du fichier identifié à la ligne L6.
- L4) Expression analytique de la fonction d'erreur en terme des variables x , y et z (notation Maple V).
- L6) Nom du fichier qui contient les données du champ scalaire servant à créer la métrique.
- L8) Nombre de fois que la séquence d'adaptation (3.2) sera exécutée.
- L10) Variable booléenne dont la valeur détermine si le déraffinement d'arêtes de la séquence d'adaptation (3.2) va être effectué ou non.
- L12) Variable booléenne `DeraffineSommet1D` dont la valeur détermine si le déraffinement d'arêtes (qui est, en fait, un déraffinement de sommets) va permettre d'enlever des sommets du bord (i.e. 1D). À noter que les sommets qui sont des coins (0D) ne sont jamais enlevés.
- L14) Longueur désirée des arêtes dans la métrique, notée l_d^m , définie à la section 2.2.6.
- L16) Paramètre d'initialisation de la variable l_c^m définie à la section 2.2.6. S'il vaut 1, on fixe l_c^m avec la valeur entrée par l'utilisateur au terminal, sinon on utilise la formule (2.9).
- L18) Nombre de divisions n_{div} tel que défini à la section 2.2.6.
- L20) Longueur minimale euclidienne l_{min}^e (critère 2.11) tolérée dans le maillage. Les arêtes plus petites que cette longueur seront enlevées.
- L22) Longueur maximale euclidienne l_{max}^e (critère 2.11) tolérée dans le maillage. Les arêtes plus grandes que cette longueur seront coupées en deux.

4.4 Validation du code

Dans la section qui suit, nous allons vérifier que les algorithmes décrits au chapitre 3 ont bel et bien le comportement prévu. Pour ce faire, nous allons faire varier certains des

paramètres du fichier `testAdaptation2DMetriqueChampScalaireUsager.txt` (Sect. 4.3.3) et modifier la séquence d'adaptation (Sect 3.5).

4.4.1 Géométrie et maillage de départ

Le domaine géométrique utilisé est un triangle et il est illustré à la figure 4.1. À la figure 4.2, nous voyons le maillage de départ (noté ORIGINAL) à partir duquel tous les tests vont être faits. Les données relatives à ce maillage sont fournies au tableau 4.2. On se donne aussi une forme analytique de la fonction d'erreur à évaluer en chaque sommet du maillage pour le calcul de la métrique (Sect. 2.2.2). Elle est définie par

$$S = \frac{1}{(x^2 + y^2)^{\frac{1}{8}} + 0.08}. \quad (4.1)$$

Une représentation graphique de cette fonction sur le maillage original est donnée à la figure 4.3. De même sur le graphe illustré à la figure 4.4, la cote représente la métrique générée à partir de S .

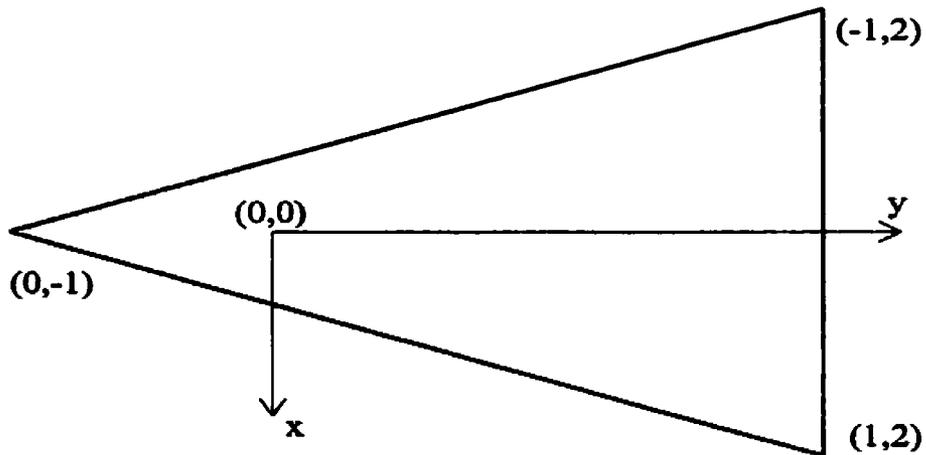


FIG. 4.1 – Géométrie utilisée pour la validation.

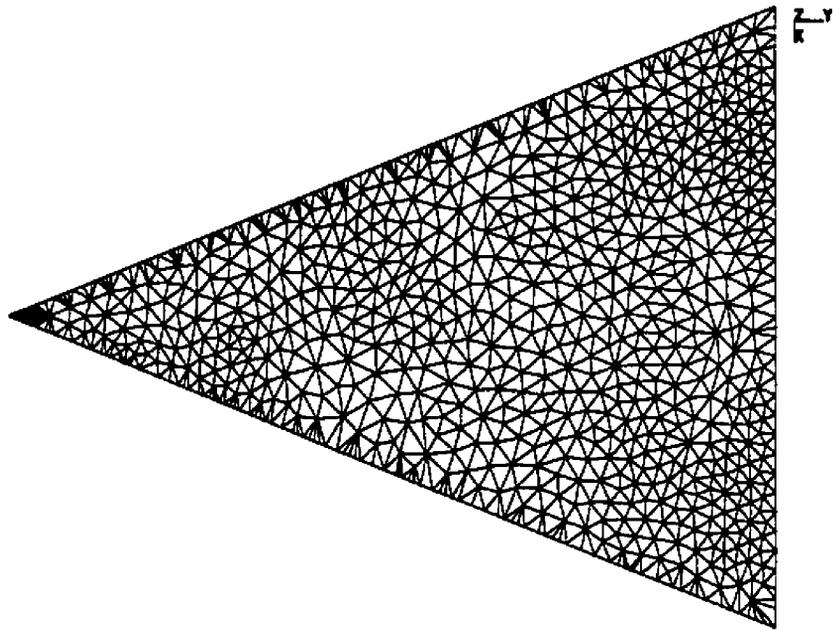


FIG. 4.2 – Maillage de base (nommé ORIGINAL) utilisé pour la validation.

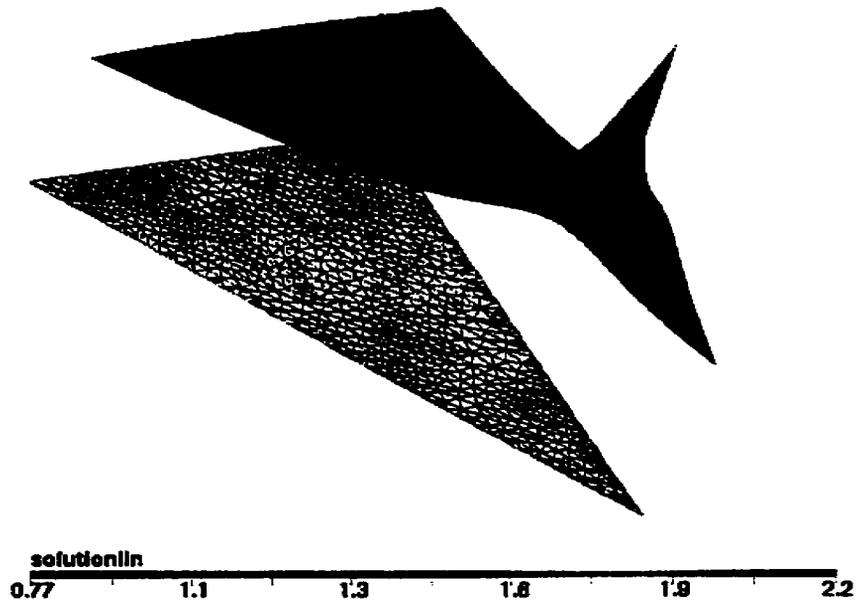


FIG. 4.3 – Fonction analytique de l'erreur (4.1).

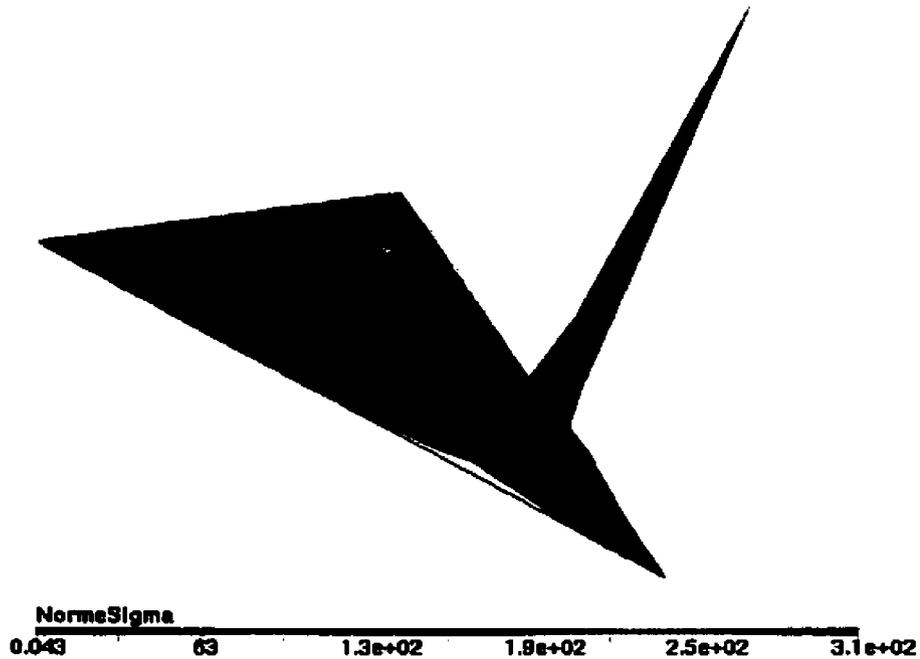


FIG. 4.4 – Norme de la métrique.

4.4.2 Adaptation de référence et fichier de paramètres

Nous allons maintenant faire une boucle d'adaptation en utilisant les paramètres que l'on retrouve dans le tableau 4.1. La séquence d'adaptation sera la suivante

$$\text{Séquence d'adaptation} \left\{ \begin{array}{l} 1. \text{ Raffinement d'arêtes} \\ 2. \text{ Séquence de nettoyage} \\ 3. \text{ Déraffinement d'arêtes} \\ 4. \text{ Séquence de nettoyage} \\ 5. \text{ Séquence de nettoyage} \end{array} \right. \quad (4.2)$$

Le maillage obtenu servira de référence pour comparer l'effet des différents paramètres et sera noté RÉFÉRENCE. Il est illustré à la figure 4.5. On peut voir un agrandissement du voisinage de $(0,0)$, où l'erreur est la plus grande, à la figure 4.6. Cette première séquence d'adaptation nous permet de constater que le raffinement et le déraffinement d'arêtes fonctionnent bien, puisque beaucoup de sommets ont été créés là où l'erreur est grande et plusieurs ont été enlevés là où l'erreur est petite.

Les tests subséquents ont pour but de vérifier que ce choix de paramètres et d'ordre des opérations sur les maillages n'est pas arbitraire. Pour le montrer, nous ferons varier les paramètres des lignes L14 et L20. Les nouvelles valeurs de ces deux paramètres, qui correspondent à chacune des variations, apparaissent au tableau 4.3. Les caractéristiques des maillages qui seront générés dans cette section sont indiquées au tableau 4.2. On peut y voir que la meilleure qualité moyenne des éléments correspond au maillage RÉFÉRENCE.

Dans le but de résoudre les problèmes d'éléments finis du chapitre 5, nous utiliserons ces paramètres et nous nous baserons sur cette séquence pour construire la séquence d'adaptation 3.2. Bien que nous n'ayons pas encore énoncé ce problème, pour comprendre les remarques suivantes, il suffit de savoir que la variable de contrôle sera nommé σ et sera la métrique construite à partir de la fonction analytique 4.1. Nous pouvons voir à la figure 4.7 la norme de σ calculée sur le maillage de référence et, à la figure 4.8, la même norme mais avec un maillage de référence modifié pour introduire des éléments plats. La différence obtenue entre les métriques est remarquable. Cela montre que les éléments plats ont un effet néfaste sur le calcul de la métrique. Cependant nous devons tolérer un certain étirement des triangles, sans diminuer de façon importante la qualité des éléments. En outre, nous observons sur la figure 4.8 un certain nombre d'éléments dont la qualité est de l'ordre de 10^{-10} . Cette qualité devrait donc être améliorée. Or, l'application du critère de longueur minimale nous interdirait un retournement car l'amélioration de la qualité d'un triangle avait pour effet de diminuer la longueur d'une arête déjà trop petite. Il y a donc un conflit entre deux besoins et dans ce cas, il vaut mieux ignorer ce critère. Certaines validations concernant cette remarque seront détaillées à la section 4.4.5.

4.4.3 Validation du déraffinement et des boucles de nettoyage

Comme premier test, nous allons changer l_d^m pour la valeur 1.0 tel qu'indiqué au tableau 4.3. Le maillage déraffiné (noté DÉRAF1) est illustré à la figure 4.9 et les données correspondantes sont inscrites dans le tableau 4.2. On voit que plus de 50% des sommets ont été enlevés du maillage ORIGINAL, ce qui correspond parfaitement au résultat

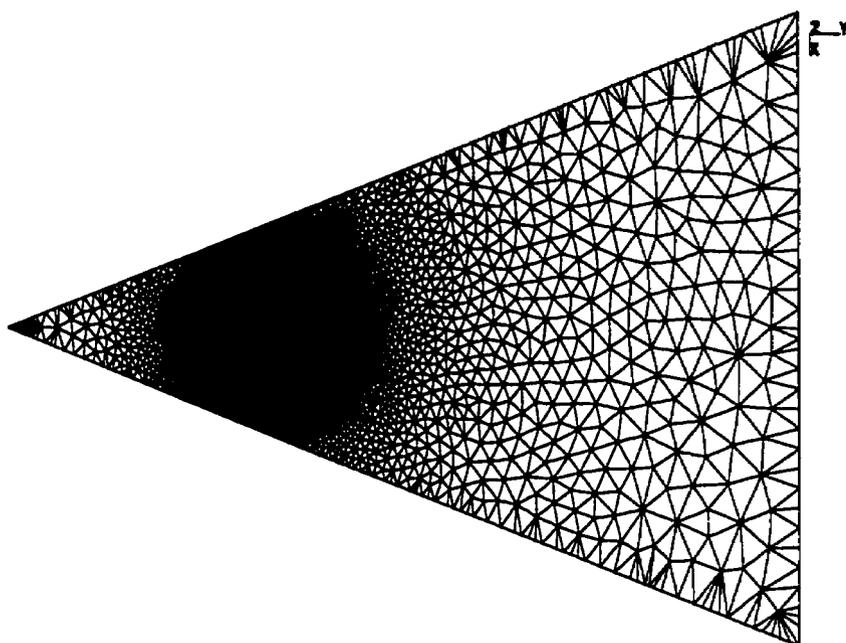


FIG. 4.5 – Maillage RÉFÉRENCE obtenu avec l'adaptation dite de référence.

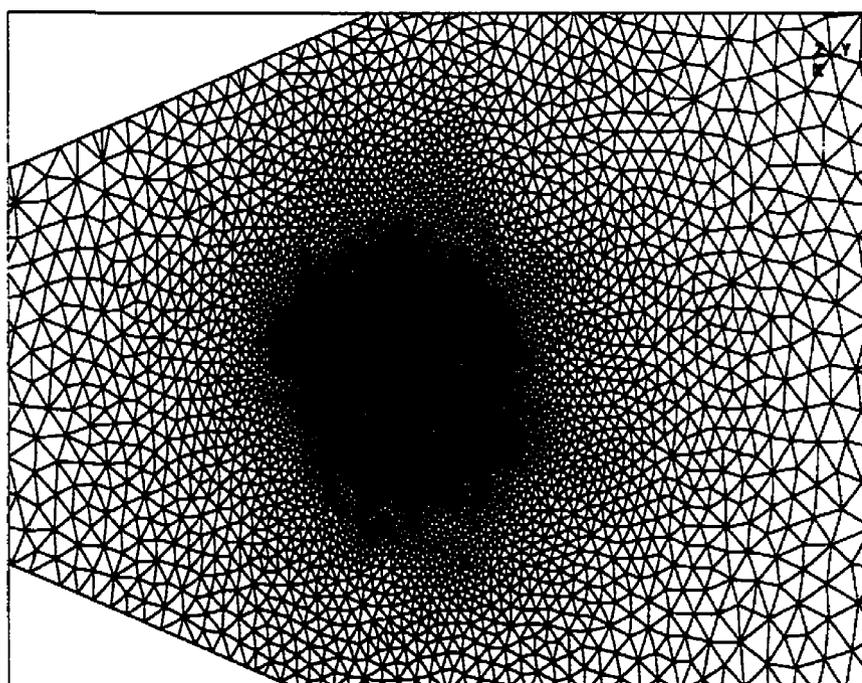


FIG. 4.6 – Agrandissement du maillage RÉFÉRENCE.

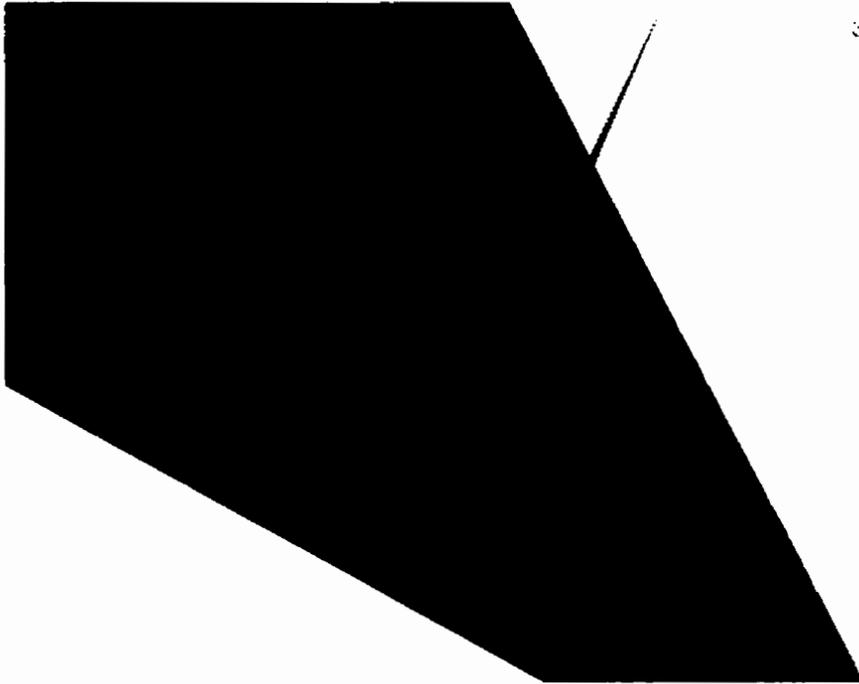


FIG. 4.7 – Norme de σ calculée sur le maillage de référence.

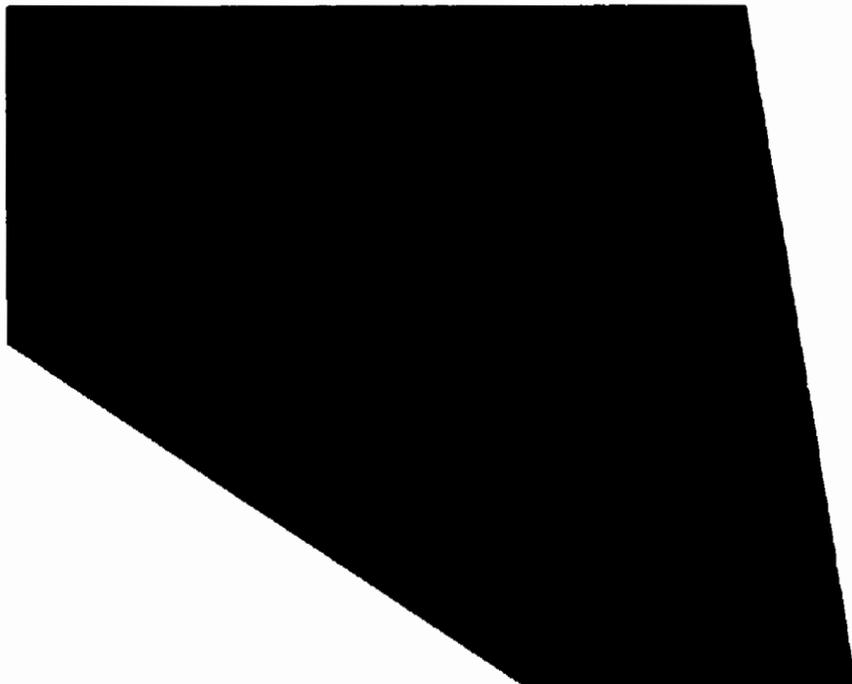


FIG. 4.8 – Norme de σ calculée sur un maillage contenant des éléments très plats.

Maillage	Nombre de sommets	Nombre d'arêtes	Nombre d'éléments	Qualité moyenne des éléments
ORIGINAL	867	2364	1498	0.834188
RÉFÉRENCE	4579	13474	8896	0.884485
DÉRAF1	427	1044	618	0.635997
DÉRAF2	427	1044	618	0.652373
RAFDÉRAF	4366	12844	8479	0.599501
RAFFINEMENT	3110	9071	5962	0.845961
INVERSION	6970	20646	13677	0.767461

TAB. 4.2 – Données des maillages pour les tests de validation.

Maillage	Ligne L14 (l_d^m)	Ligne L20 (l_{min}^e)
RÉFÉRENCE	0.01	0.005
DÉRAF1	1.0	0.005
DÉRAF2	1.0	0.005
RAFDÉRAF	0.01	0.005
RAFFINEMENT	0.01	0.01
INVERSION	0.01	0.005

TAB. 4.3 – Fichier de paramètres d'adaptation pour la validation.

escompté. Par contre, l'agrandissement (Fig. 4.10) du coin supérieur droit montre que certains triangles ont trois sommets sur le bord. Nous voudrions maintenant montrer que les protections mises dans l'algorithme de retournement d'arêtes dans le but d'enrayer ce type de triangle fonctionnent bel et bien. S'il ne se basait que sur le critère de la qualité des éléments, l'algorithme ne retournerait pas l'arête opposée au coin car la qualité des nouveaux triangles obtenus serait grandement diminuée. Mais, en forçant une séquence de nettoyage de plus à la boucle d'adaptation 4.2, on obtient un maillage (noté DÉRAF2) qui ne possède plus ce type de triangle. On peut voir un agrandissement du coin supérieur droit à la figure 4.11 qui illustre bien ce fait. Il faut donc noter qu'on doit effectuer plusieurs séquences de nettoyage avant d'obtenir un maillage adéquat. C'est pourquoi, on va effectuer deux séquences de nettoyage pour terminer la séquence d'adaptation pour minimiser l'occurrence de ces configurations. De plus, on voit que la qualité moyenne des éléments est passée de 0.636 à 0.652 ce qui démontre que le déplacement de sommets et le retournement d'arêtes améliorent la qualité des éléments. De ces conclusions, nous allons ajouter une séquence de nettoyage pour porter à trois le nombre de séquences de nettoyage qui terminent l'adaptation.

4.4.4 Validation de l'amélioration de la qualité des éléments

Comme nous l'avons fait remarquer à la section précédente, la séquence de nettoyage améliore la qualité des éléments. Pour valider cette idée et bien mettre en évidence l'effet de la séquence de nettoyage, nous allons effectuer un test qui consiste à enlever toutes les opérations de déplacement de sommets et de retournement d'arêtes dans la séquence d'adaptation 4.2. Elle se réduit ainsi : un raffinement suivi d'un déraffinement d'arêtes. Un grossissement aux alentours du point (0,0) du maillage obtenu (noté RAFDÉRAF) est illustré à la figure 4.12. On voit clairement que la qualité moyenne des éléments est nettement inférieure à celle du maillage RÉFÉRENCE. En effet, elle n'est que de 0.6 comparée à 0.88 pour le maillage RÉFÉRENCE, ce que l'on distingue facilement à l'oeil nu. Par contre, le nombre de sommets est inférieur de plus de 200 sommets. Cela est dû aux séquences de nettoyage qui sont effectuées entre le raffinement et le déraffinement d'arêtes.

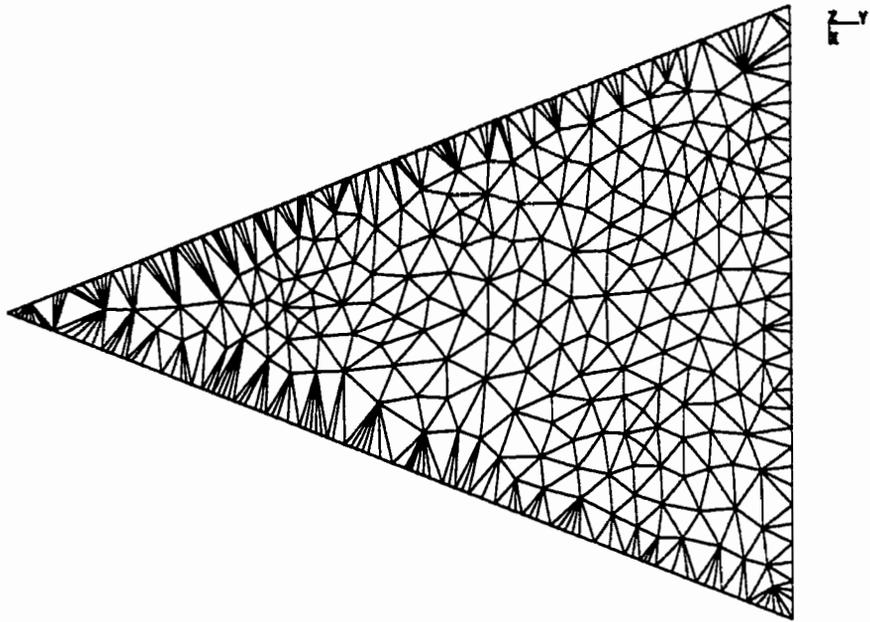


FIG. 4.9 – Maillage DÉRAF1 obtenu pour la validation du déraffinement.

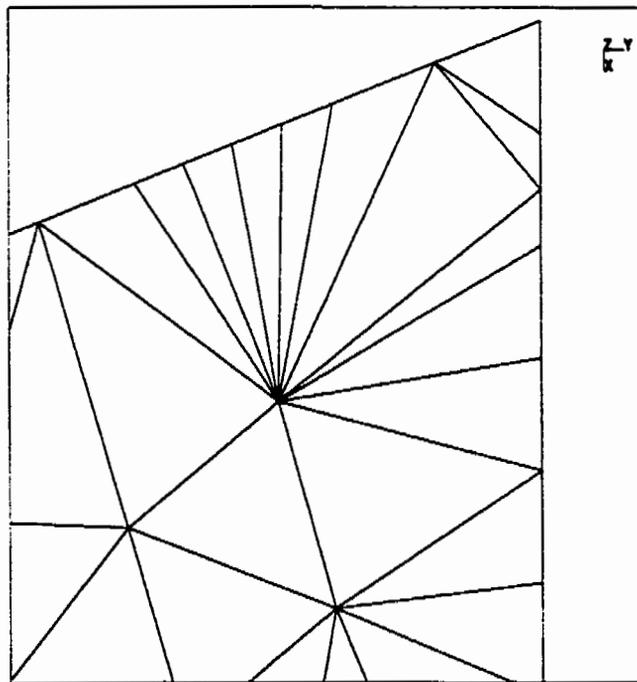


FIG. 4.10 – Agrandissement du maillage DÉRAF1, coin supérieur droit.

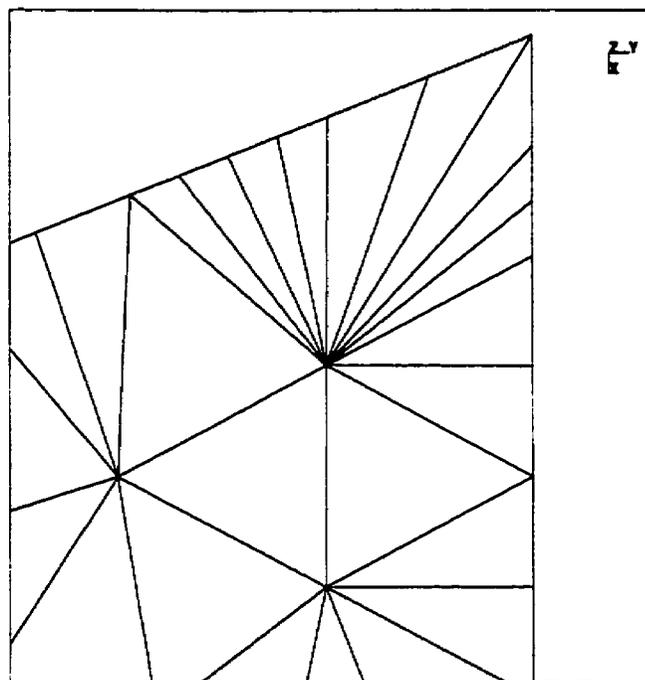


FIG. 4.11 – Agrandissement du maillage DÉRAF2, coin supérieur droit.

Ces opérations supplémentaires ont pour effet de rallonger ou de raccourcir la longueur de certaines arêtes dans la métrique. Le nombre inférieur de sommets s'explique par l'augmentation de la longueur des arêtes trop courtes dans la métrique dû aux séquences de nettoyage et par le fait même, à la diminution du nombre d'arêtes déraffinées.

4.4.5 Validation du critère de la longueur euclidienne minimale

Nous devons aussi vérifier que les opérations d'adaptation respectent le critère 2.11 et plus particulièrement que le raffinement d'arêtes ne crée pas des arêtes dont la longueur euclidienne est inférieure à l_{min}^e . Pour ce faire, nous modifions le paramètre de la ligne L20 pour le fixer à 0.01 au lieu de 0.005. L'effet obtenu est clair, on a une diminution de 1469 sommets par rapport au maillage RÉFÉRENCE, ce qui équivaut à une diminution du tiers du nombre de sommets. À la figure 4.13, on voit un agrandissement de ce maillage obtenu (noté RAFFINEMENT) aux alentours de (0,0). On y remarque qu'en général les arêtes sont plus longues que celles du maillage de référence, mais sensiblement de la

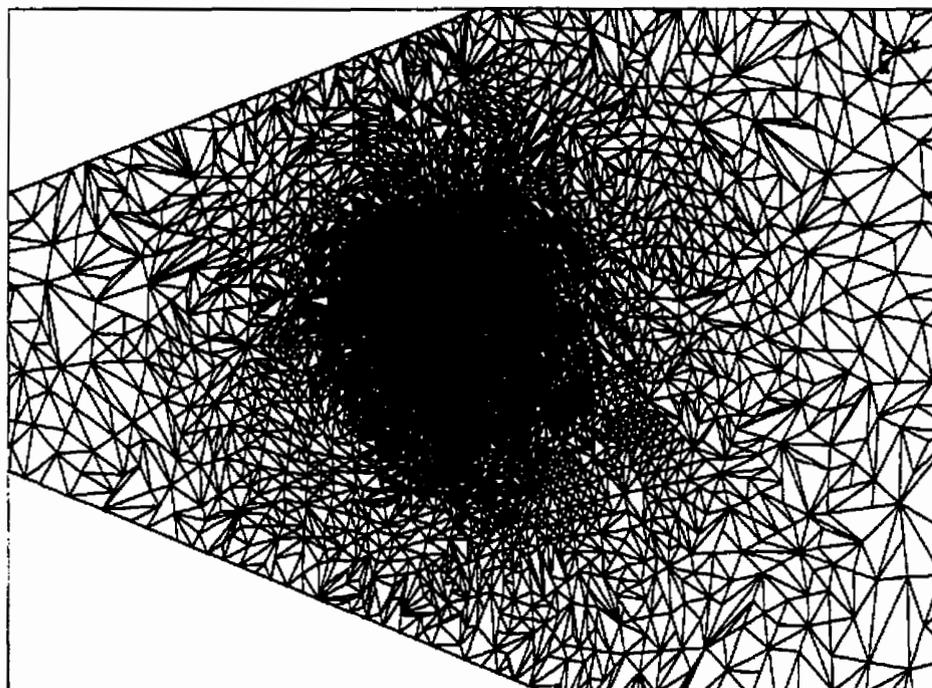


FIG. 4.12 – Agrandissement du maillage RAFDÉRAF obtenu en enlevant les séquences de nettoyage.

même longueur aux alentours de $(0,0)$. Ceci montre que le critère est très performant dans le cas du raffinement. Si nous revenons brièvement sur la question de l'application de ce critère à l'opération de retournement, nous constatons de nouveau l'apparition de plusieurs triangles aplatis que nous ne pouvons pas retourner. Cela vient du fait que la longueur euclidienne l^e avant le retournement (l_{avant}^e) et après ($l_{\text{après}}^e$) ne satisfont pas (2.11) et que $l_{\text{avant}}^e > l_{\text{après}}^e$. On peut voir un exemple de ce cas avec la configuration illustrée à la figure 4.14. Cette arête n'a pu être retournée parce que le critère précédent n'était pas satisfait. Le résultat est montré à la figure 4.15. La qualité minimale des éléments du maillage est passée de 3.4385×10^{-10} à 1.351×10^{-4} . On a le même raisonnement si $l^e > l_{\text{max}}^e$.

4.4.6 Validation de la séquence d'adaptation

On en vient maintenant à se demander si le raffinement d'arêtes doit être fait avant

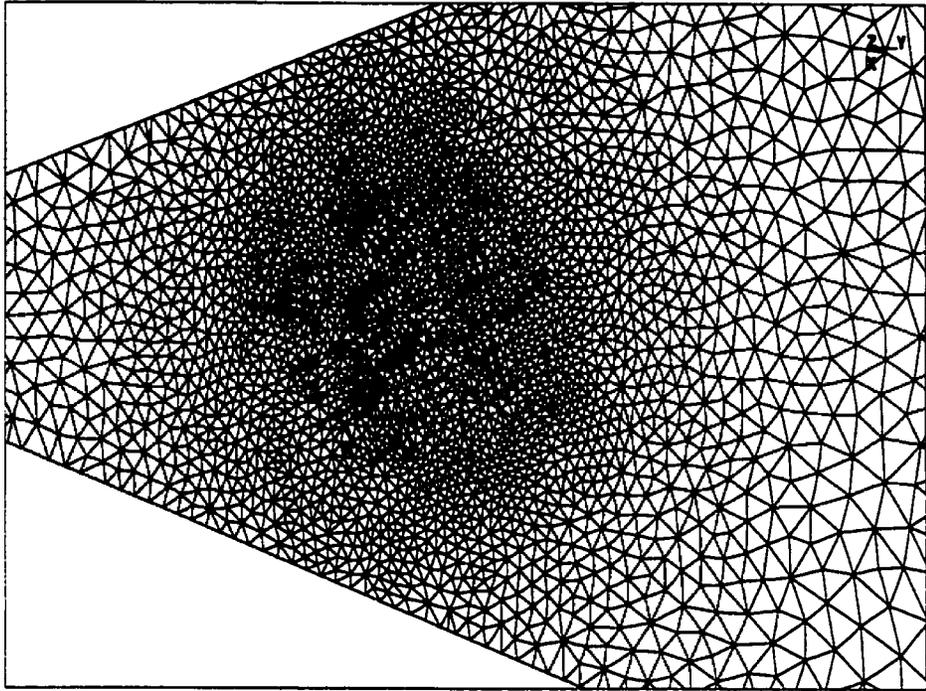


FIG. 4.13 – Agrandissement du maillage RAFFINEMENT obtenu en posant l_{min}^e assez grand.

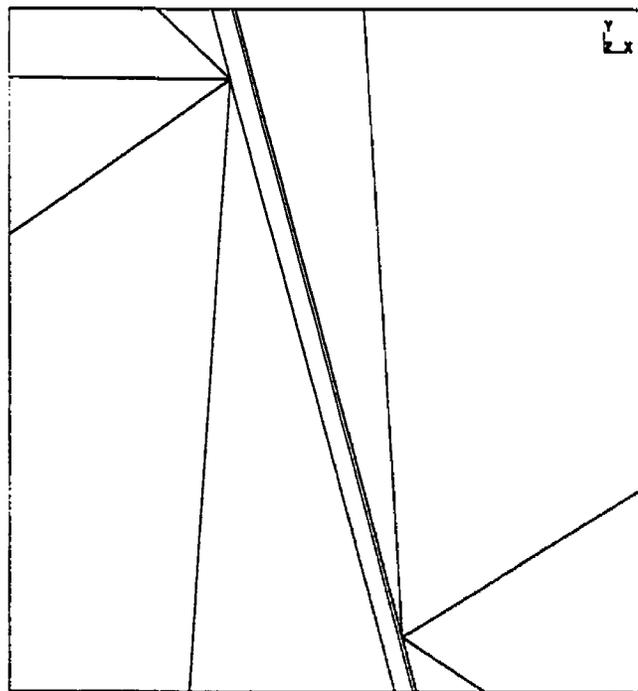


FIG. 4.14 – Agrandissement du maillage TestCritereLongEuclMinAvantRet.

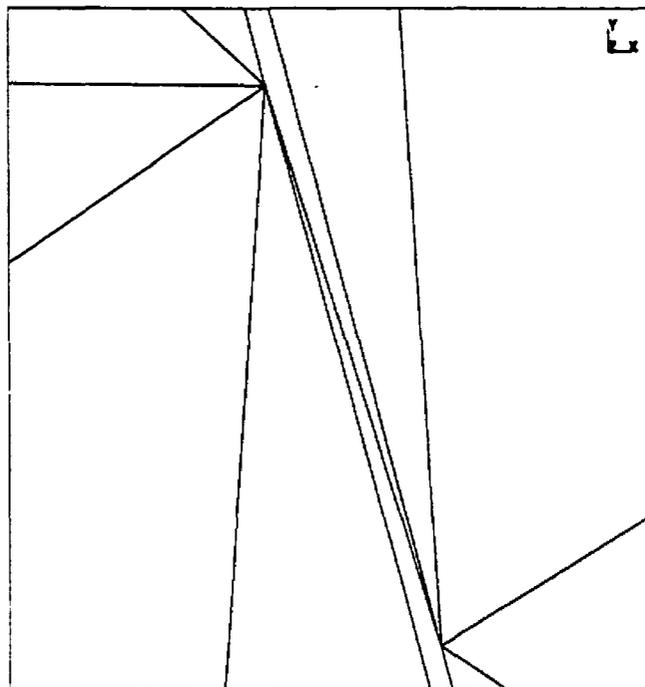


FIG. 4.15 – Agrandissement du maillage TestCritereLongEuclMinApresRet.

le déraffinement. Intuitivement, on est amené à penser que oui, puisque la métrique est définie sur les sommets du maillage. Or, débiter par un déraffinement aura pour effet de réduire le nombre de sommets et, par le fait même, de diminuer la précision de l'estimateur d'erreur. Nous avons donc effectué ce test en inversant le raffinement et le déraffinement dans la séquence d'adaptation 4.2. En observant la figure 4.16, qui est un agrandissement du maillage adapté (noté INVERSION), on voit bien que l'effet décrit ci-haut se produit. De plus, le nombre de sommets diffère d'environ 2400 sommets soit une augmentation de plus de 50% des sommets comparativement au maillage RÉFÉRENCE.

Nous pouvons aussi nous questionner sur la nécessité de l'opération de nettoyage effectuée entre le raffinement et le déraffinement d'arêtes. Pour ce faire, nous allons expérimenter la séquence (4.3) et comparer avec la séquence (4.2). Nous allons comparer leurs effets sur deux problèmes du chapitre 5 qui sont le problème de Poiseuille avec $\lambda = 1.0$ (voir Sect. 5.5.3) et le problème de la couche limite avec $\lambda = 0.0$ (voir Sect. 5.5.5). Ce test a pour seul but de comparer les maillages obtenus et non les solutions correspondantes. Il

n'est donc pas utile de connaître les problème eux-mêmes puisque nous ne voulons nous concentrer que sur la qualité des éléments. Les résultats concernant le premier problème sont détaillés aux tableaux 4.4 et 4.5 ainsi que le deuxième problème aux tableaux 4.6 et 4.7.

	Maillages		
Erreur	ORIGINAL	ADAPT1	ADAPT2
Nb. Éléments	938	1500	1401
Nb. Sommets	510	827	790
Qualité Min.	0.708772	0.261922	0.213188
Qualité Max.	0.999993	0.999471	0.999925
Qualité Moy.	0.968861	0.824974	0.81758

TAB. 4.4 – Résultats pour le problème de Poiseuille : Cas $\lambda = 1.0$. Adaptation effectuée sans opération intermédiaire de nettoyage.

	Maillages		
Erreur	ORIGINAL	ADAPT1	ADAPT2
Nb. Éléments	938	1449	1354
Nb. Sommets	510	822	795
Qualité Min.	0.708772	0.301212	0.0360937
Qualité Max.	0.999993	0.999888	0.999438
Qualité Moy.	0.968861	0.809698	0.805321

TAB. 4.5 – Résultats du problème de Poiseuille : Cas $\lambda = 1.0$. Adaptation effectuée avec une opération intermédiaire de nettoyage.

$$\text{Séquence d'adaptation} \left\{ \begin{array}{l} 1. \text{ Raffinement d'arêtes} \\ 2. \text{ Déraffinement d'arêtes} \\ 3. \text{ Séquence de nettoyage} \\ 4. \text{ Séquence de nettoyage} \end{array} \right. \quad (4.3)$$

	Maillages			
Erreur	ORIGINAL	ADAPT1	ADAPT2	ADAPT3
Nb. Éléments	938	5225	7555	7343
Nb. Sommets	510	2798	4180	4206
Qualité Min.	0.708772	0.117188	0.0710509	0.0506843
Qualité Max.	0.999993	0.999996	0.9999	0.99999
Qualité Moy.	0.968861	0.845665	0.817988	0.801485

TAB. 4.6 – Résultats du problème de la couche limite : Cas $\lambda = 0.0$. Adaptation effectuée sans opération intermédiaire de nettoyage.

	Maillages			
Erreur	ORIGINAL	ADAPT1	ADAPT2	ADAPT3
Nb. Éléments	938	4989	7644	7877
Nb. Sommets	510	2707	4290	4551
Qualité Min.	0.708772	0.162404	0.0807741	0.0664856
Qualité Max.	0.999993	0.999958	0.999987	0.999958
Qualité Moy.	0.968861	0.86136	0.826698	0.811404

TAB. 4.7 – Résultats du problème de la couche limite : Cas $\lambda = 0.0$. Adaptation effectuée avec une opération intermédiaire de nettoyage.

L'analyse des tableaux correspondants, ne permet pas de tirer de conclusion sur la nécessité de la séquence de nettoyage placée entre le raffinement et le déraffinement. Comme nous devons faire un choix, nous avons préféré la séquence (4.2).

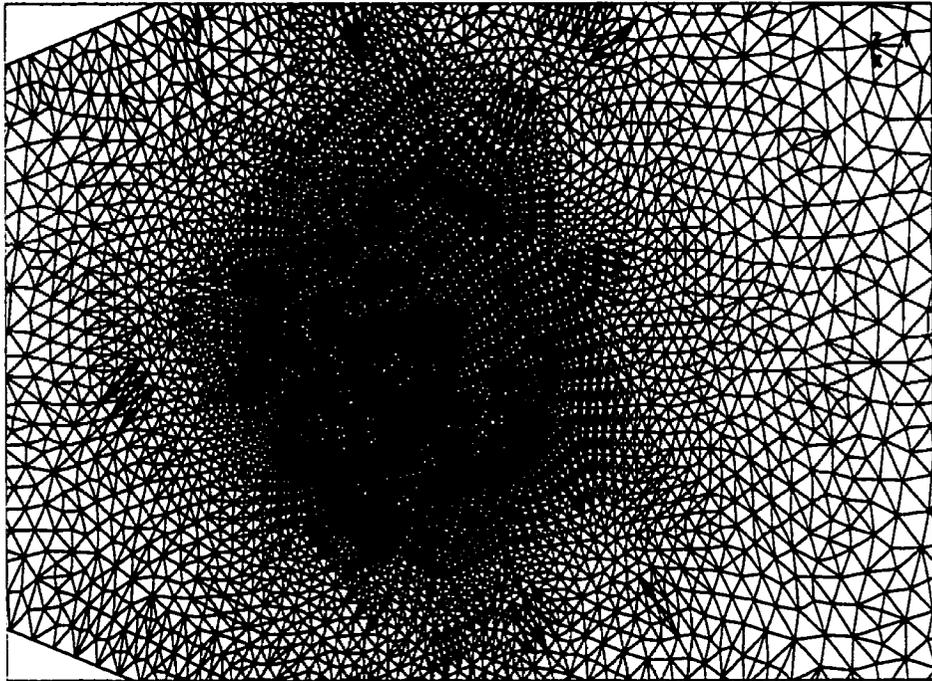


FIG. 4.16 – Agrandissement du maillage INVERSION obtenu en inversant raffinement et déraffinement.

4.4.7 Conclusion sur les tests

En conclusion, on voit que les opérations d'adaptation ont bien le comportement désiré. De plus, le maillage de référence obtenu est régulier et bien adapté en fonction de la métrique de l'erreur. Cette section nous a permis d'expérimenter l'effet des paramètres d'adaptation et d'en tirer une séquence d'adaptation convenable, définie par (3.2) et un ensemble de paramètres (tableau 4.1) qui serviront dans le chapitre 5 pour résoudre des problèmes de mécanique des fluides.

Chapitre 5

Applications aux fluides viscoélastiques

Dans ce chapitre, nous allons appliquer les techniques d'adaptation de maillages introduites aux chapitres précédents dans le contexte de la simulation numérique d'écoulements de fluides viscoélastiques. Le chapitre débutera par une brève présentation des principes généraux de la mécanique des fluides. Ensuite, nous étudierons une classe particulière de fluides viscoélastiques. Finalement, nous présenterons des résultats d'adaptation sur des problèmes tests d'écoulements.

5.1 Les principes fondamentaux de la mécanique des fluides

5.1.1 La loi de conservation de la masse et les fluides incompressibles

L'équation de continuité

Notons par $\Omega \subset \mathbb{R}^2$ un ouvert de frontière Γ suffisamment régulière formant le domaine d'écoulement du fluide.

Une des lois fondamentales de la physique stipule qu'il n'y a aucune création ou destruction de matière dans l'univers. Cette loi de conservation de la masse se traduit par l'équation mathématique suivante

$$\frac{d}{dt} \int_{W(t)} \rho(t, \mathbf{x}) \, d\mathbf{x} = 0, \quad (5.1)$$

où $\rho(t, \mathbf{x})$ est la densité du fluide en un point \mathbf{x} au temps t et $W(t)$ est un sous-domaine de Ω qui peut se déplacer et se déformer par l'action du fluide en fonction du temps. L'équation traduit donc l'absence de variation de la masse d'un sous-domaine du fluide en fonction du temps.

Si \mathbf{u} dénote le champ de vitesse du fluide, la trajectoire d'une particule fluide initialement située au point \mathbf{x} au temps $t = 0$, est obtenue en intégrant le système d'équations différentielles

$$\begin{cases} \frac{d\mathbf{x}(t)}{dt} = \mathbf{u}(t, \mathbf{x}(t)), \\ \mathbf{x}(0) = \mathbf{x}. \end{cases}$$

Notons par $\Phi(t, \mathbf{x})$ la solution du système ci-dessus qui dépend de la condition initiale \mathbf{x} . La fonction Φ porte le nom de flot du champ de vecteurs \mathbf{u} . Le changement de variable $\mathbf{x} = \Phi(t, \mathbf{X})$ représente le point de vue matériel. Donc regarder une quantité du point de vue matériel signifie qu'on la regarde comme fonction du temps et de la position initiale du fluide.

Étant donnée une fonction $f(t, \mathbf{x})$ exprimée en coordonnées spatiales, on peut appliquer le changement de variables pour obtenir la fonction $F(t, \mathbf{X}) = f(t, \Phi(t, \mathbf{X}))$ définie par rapport aux coordonnées matérielles. Nous pouvons par la suite introduire la notion de **dérivée matérielle** ($\frac{D}{Dt}$) d'une fonction scalaire f . La dérivée matérielle est définie par

$$\frac{Df}{Dt}(t, \mathbf{x}) = \frac{d}{dt} f(t, \Phi(t, \mathbf{x})).$$

En utilisant la définition du flot Φ , on obtient l'expression suivante pour la dérivée matérielle

$$\frac{Df}{Dt} = \frac{\partial f}{\partial t} + \nabla f \cdot \mathbf{u}.$$

De plus, cet opérateur possède les propriétés suivantes :

$$1. \frac{D}{Dt}(f + g) = \frac{Df}{Dt} + \frac{Dg}{Dt}$$

$$2. \frac{D}{Dt}(f \cdot g) = f \frac{Dg}{Dt} + g \frac{Df}{Dt}$$

On peut consulter [10] pour plus de détails. Le théorème suivant permet d'obtenir une autre forme pour l'équation (5.1).

Théorème 3 *Le principe de conservation de la masse décrit par (5.1) est équivalent à l'équation locale suivante*

$$\frac{D\rho}{Dt} + \rho \nabla \cdot \mathbf{u} = 0. \quad (5.2)$$

Cette dernière équation sera nommée équation de continuité.

Les fluides incompressibles

En général, sous l'influence de la pression, les liquides ne changent pas de volume. Un tel fluide est dit incompressible. Si la densité $\rho(t, \mathbf{x})$ est partout constante (fluide homogène), il y a équivalence entre l'incompressibilité et la loi de conservation de la masse. En effet, il suffit de remplacer $\rho = \rho_0$ dans l'équation (5.2) et on en déduit le théorème suivant :

Théorème 4 *Si la densité est constante, un fluide satisfait le principe de conservation de la masse si et seulement si*

$$\nabla \cdot \mathbf{u} = 0. \quad (5.3)$$

5.1.2 La loi de conservation de la quantité de mouvement

Le principe des contraintes de Cauchy

Un autre grand principe en physique est la **conservation de la quantité de mouvement**. Les fluides n'échappent évidemment pas à cette loi. Si l'on considère une partie $W \subset \Omega$ du fluide de frontière S à l'instant $t = 0$ et $W_t = \Phi(t, W)$ la partie convectée par le fluide, alors la variation de la quantité de mouvement contenu dans W_t est due à la somme des forces externes (comme la gravité) et à la somme des forces internes (comme le

frottement dû à la viscosité) agissant sur la frontière de la région convectée. Si \mathbf{f} symbolise les forces externes par unité de masse et \mathbf{t} les forces internes par unité de surface (où \mathbf{n} est la normale à la surface), alors on obtient l'équation vectorielle suivante :

$$\frac{d}{dt} \int_{W_t} \rho \mathbf{u} \, d\mathbf{x} = \int_{W_t} \rho \mathbf{f} \, d\mathbf{x} + \int_{S_t} \mathbf{t} \cdot \mathbf{n} \, da \quad (5.4)$$

Voici maintenant un important théorème que l'on doit à Cauchy concernant les forces internes dues au milieu continu.

Théorème 5 *La loi de conservation de la quantité de mouvement implique qu'il existe une fonction à valeur matricielle $\sigma_{ij}(t, \mathbf{x})$ telle que $\mathbf{t}(t, \mathbf{x}, \mathbf{n}) = \boldsymbol{\sigma}(t, \mathbf{x}) \cdot \mathbf{n}$.*

Notons que $\boldsymbol{\sigma}$ sera appelé le tenseur des contraintes.

L'équation du mouvement

Nous allons introduire trois outils fondamentaux qui sont utilisés pour obtenir les formes locales des lois de conservation. On trouvera les démonstrations dans les ouvrages de références [10] et [11].

Théorème 6 (Théorème de Gauss) *Soit un champ de vecteurs \mathbf{u} de classe C^k où $k \geq 1$, le flux de \mathbf{u} à travers une surface fermée S dont la normale \mathbf{n} pointe vers l'extérieur est égal à l'intégrale de la divergence de \mathbf{u} sur le volume W limité par S .*

$$\int_W \nabla \cdot \mathbf{u} \, d\mathbf{x} = \oint_S \mathbf{u} \cdot \mathbf{n} \, dS \quad (5.5)$$

Théorème 7 (Le principe de localisation) *Soit une fonction f continue définie sur un domaine $\Omega \subset \mathbb{R}^n$. Si $\int_W f \, d\mathbf{x} = 0$ pour tout $W \subset \Omega$ alors $f \equiv 0$ sur Ω .*

Théorème 8 (Le théorème de transport) *Soit un champ de vecteurs $\mathbf{u}(t, \mathbf{x})$ de classe C^2 sur le domaine Ω et parallèle au bord $\partial\Omega$ ($\mathbf{u} \cdot \mathbf{n} = 0$ sur $\partial\Omega$). Supposons que le flot $\Phi(t, \mathbf{x})$ de \mathbf{u} est inversible pour tout \mathbf{x} et pour un certain intervalle de t . Si $f(t, \mathbf{x})$ est*

une fonction définie sur Ω , nous avons sur cet intervalle de t

$$\frac{d}{dt} \int_{\Phi_t(W)} f(t, \mathbf{x}) d\mathbf{x} = \int_{\Phi_t(W)} \left(\frac{\partial f}{\partial t} + \mathbf{u} \cdot \nabla f + f \nabla \cdot \mathbf{u} \right) d\mathbf{x} = \int_{\Phi_t(W)} \left(\frac{Df}{Dt} + f \nabla \cdot \mathbf{u} \right) d\mathbf{x}$$

où W est un sous-ensemble de Ω possédant une frontière continue par morceaux.

Nous allons maintenant appliquer le principe de localisation à l'équation de la conservation de la quantité de mouvement. À l'aide du théorème de transport et du théorème de Gauss, nous pouvons écrire

$$0 = \int_W \left(\frac{D\rho u_i}{Dt} + \rho u_i \nabla \cdot \mathbf{u} - \rho f_i - (\nabla \cdot \boldsymbol{\sigma})_i \right) dx, \quad i = 1, 2 \quad (5.6)$$

où $(\nabla \cdot \boldsymbol{\sigma})_i = \sum_{j=1}^3 \frac{\partial \sigma_{ij}}{\partial x_j}$. En appliquant le principe de localisation et les propriétés de la dérivée matérielle, on peut supprimer le signe d'intégration et obtenir l'équation vectorielle

$$\rho \frac{D\mathbf{u}}{Dt} + \mathbf{u} \frac{D\rho}{Dt} + \rho \mathbf{u} \nabla \cdot \mathbf{u} - \rho \mathbf{f} - \nabla \cdot \boldsymbol{\sigma} = 0. \quad (5.7)$$

Si le principe de la conservation de la masse est vérifié (équation de la continuité), nous avons

$$\frac{D\rho}{Dt} + \rho \nabla \cdot \mathbf{u} = 0.$$

En substituant dans la forme locale de la conservation de la quantité de mouvement (5.7), on obtient l'équation du mouvement, connue aussi sous le nom de **premier principe de Cauchy**

$$\rho \frac{D\mathbf{u}}{Dt} = \nabla \cdot \boldsymbol{\sigma} + \rho \mathbf{f}. \quad (5.8)$$

Une propriété intéressante du tenseur des contraintes $\boldsymbol{\sigma}$ est déduite si on a, en plus de la conservation de la quantité de mouvement, la conservation du moment angulaire. L'équation qui traduit le phénomène est la suivante :

$$\frac{d}{dt} \int_{W_t} \rho(\mathbf{x} \times \mathbf{u}) d\mathbf{x} = \int_{W_t} \rho(\mathbf{x} \times \mathbf{f}) d\mathbf{x} + \int_{S_t} \mathbf{x} \times \mathbf{t} da, \quad (5.9)$$

où le terme de gauche traduit le taux de variation du moment angulaire. Alors nous pouvons écrire le résultat suivant bien connu sous le nom de **deuxième principe de Cauchy** :

Théorème 9 *Si la conservation de la quantité de mouvement (5.4) et l'équation de continuité (5.2) sont respectées, alors on a la conservation du moment angulaire si et seulement si le tenseur des contraintes est symétrique.*

Nous allons donc aborder l'étude des fluides en utilisant les théorèmes mentionnés ci-haut.

5.2 Le système de Stokes pour les fluides incompressibles

Dans le cas des fluides parfaits, la viscosité est ignorée et le tenseur des contraintes est $\boldsymbol{\sigma} = -p\mathbf{I}$ où p est la pression associée au système et \mathbf{I} le tenseur identité. Comme nous considérons un système plus général, nous devons tenir compte de la viscosité. Il nous faut, par conséquent, reconsidérer le tenseur des contraintes $\boldsymbol{\sigma}$. Un terme $\boldsymbol{\tau}$ devra être ajouté au terme déjà connu $-p\mathbf{I}$ pour modéliser la viscosité. Il nous faut pour cela introduire le **tenseur taux de déformation** $\dot{\boldsymbol{\gamma}}(\mathbf{u})$ qui est défini par

$$\dot{\boldsymbol{\gamma}}(\mathbf{u})_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right).$$

Pour plus de détails, le lecteur peut consulter [10].

Or, la relation entre le tenseur des contraintes et le tenseur de déformation varie selon le fluide que l'on étudie. L'étude de telles relations est en fait le sujet de recherche d'une branche de la mécanique des milieux continus que l'on nomme **théorie constitutive**. Cette théorie nous indique que la partie $\boldsymbol{\tau}$ qui modélise la viscosité est une fonction du tenseur taux de déformation. En supposant cette fonction linéaire et en tenant compte de l'incompressibilité du fluide, on obtient alors la représentation suivante

$$\boldsymbol{\tau} = 2\mu \dot{\boldsymbol{\gamma}}(\mathbf{u})$$

où μ est la constante de viscosité du fluide. Alors, nous pouvons réécrire le tenseur total des contraintes sous la forme

$$\boldsymbol{\sigma} = -p\mathbf{I} + 2\mu \dot{\boldsymbol{\gamma}}(\mathbf{u}).$$

En substituant dans l'équation du mouvement (5.8) et en négligeant le terme d'inertie $\frac{D\mathbf{u}}{Dt}$ puisque l'écoulement est supposé lent, on obtient les équations de Stokes pour les fluides incompressibles :

$$0 = \mathbf{f} + \frac{1}{\rho} \nabla \cdot \boldsymbol{\sigma} = \mathbf{f} - \frac{1}{\rho} \nabla p + 2\nu \nabla \cdot \dot{\boldsymbol{\gamma}}(\mathbf{u}) \quad (5.10)$$

où $\nu = \frac{\mu}{\rho}$. Comme le fluide est incompressible et homogène, $\rho = \text{Constante}$ et peut être incorporée à p pour ainsi disparaître des équations. Le système de Stokes apparaît donc sous la forme suivante pour les fluides incompressibles :

$$\begin{cases} -2\nu \nabla \cdot \dot{\boldsymbol{\gamma}}(\mathbf{u}) + \nabla p = \mathbf{f}, \\ \nabla \cdot \mathbf{u} = 0. \end{cases} \quad (5.11)$$

5.3 Équations d'évolution des fluides viscoélastiques

Nous allons modéliser la loi de comportement d'un fluide du type viscoélastique. Une première approche dite phénoménologique associe le comportement d'un élément de fluide à un système physique analogue. Il existe un vaste choix de systèmes possibles, mais nous allons concentrer nos efforts sur un système simple qu'on nomme le **système de Maxwell**. Ce modèle décrit un élément de fluide comme étant formé d'un ressort et d'un amortisseur disposés en série tel qu'illustré sur la figure 5.1. Le ressort modélise la partie élastique du fluide et l'amortisseur, la partie visqueuse du fluide. Pour les applications dans le domaine des polymères, la partie élastique est liée à la déformation des longues chaînes macromoléculaires tandis que la partie visqueuse est associée au solvant, dans le cas d'un polymère dilué dans une solution newtonienne.

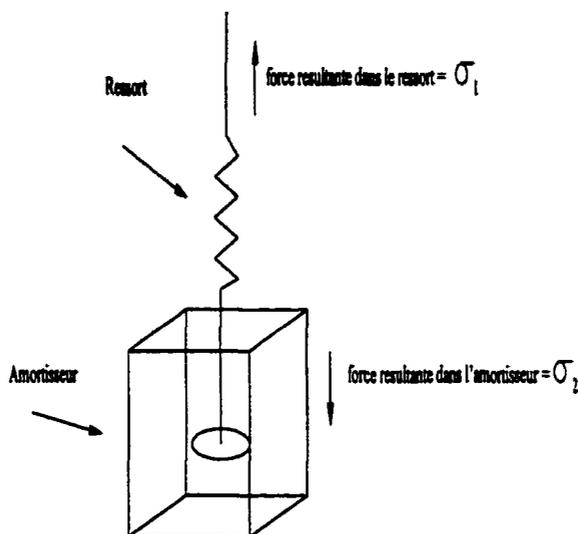


FIG. 5.1 – Représentation schématique du système physique de Maxwell.

Étudions le système décrit précédemment par des notions élémentaires de physique et supposons qu'il est unidimensionnel. Pour simplifier, supposons qu'aucune force externe ne s'applique sur le système. D'abord, la force exercée par le ressort est donnée $\sigma_1 = G\gamma_1$ où G est la constante de rappel du ressort et γ_1 le déplacement du ressort par rapport à sa position initiale.

La force exercée par l'amortisseur est donnée par $\sigma_2 = \nu \left(\frac{\partial \gamma_2}{\partial t}\right)$ où ν est le coefficient de viscosité et $\left(\frac{\partial \gamma_2}{\partial t}\right)$ la vitesse de déplacement dans l'amortisseur. Or, le fait de placer ces deux composantes en série implique que la force σ_1 est égale à σ_2 i.e. $\sigma = \sigma_1 = \sigma_2$. Nous pouvons donc trouver le taux de déplacement global

$$\frac{\partial \gamma}{\partial t} = \frac{\partial \gamma_1}{\partial t} + \frac{\partial \gamma_2}{\partial t} = \frac{(\partial \sigma / \partial t)}{G} + \frac{\sigma}{\nu}$$

qui nous amène à l'équation constitutive pour un fluide de Maxwell

$$\lambda \frac{\partial \sigma}{\partial t} + \sigma = \nu \frac{\partial \gamma}{\partial t} \tag{5.12}$$

où $\lambda = \frac{\nu}{G}$ est le temps de relaxation.

Nous allons maintenant tirer l'équation linéarisée pour le système en dimension un. Soit $S(x, t)$ la "longueur" de l'élément, $\gamma = \frac{\partial S}{\partial x}$ la déformation et $\frac{\partial S}{\partial t} = u$ la vitesse. Si nous

dérivons les deux dernières équations par rapport à t et x respectivement, nous pouvons écrire

$$\frac{\partial \gamma}{\partial t} = \frac{\partial u}{\partial x}. \quad (5.13)$$

Notons que l'équation du mouvement s'écrit

$$\rho \frac{\partial u}{\partial t} = \frac{\partial \sigma}{\partial x} \quad (5.14)$$

où σ est la force de déformation. Alors nous pouvons combiner cette dernière avec les équations (5.12) et (5.13) pour obtenir

$$\frac{\partial^2 u}{\partial t^2} + \frac{1}{\lambda} \frac{\partial u}{\partial t} = \frac{\nu}{\rho \lambda} \frac{\partial^2 u}{\partial x^2}.$$

Il est intéressant de faire varier le paramètre λ pour voir comment se comporte le fluide :

- Si l'on fait tendre le temps de relaxation vers 0 i.e. $\lambda \rightarrow 0$, le fluide est newtonien. En effet $\sigma = \nu \frac{\partial u}{\partial x}$ et $\rho \frac{\partial u}{\partial t} = \nu \frac{\partial^2 u}{\partial x^2}$ sont les équations d'un fluide newtonien.
- Par contre lorsque $\lambda \rightarrow \infty$, le fluide se comporte comme un fluide élastique : $\frac{\partial \sigma}{\partial t} = G \frac{\partial u}{\partial x}$ i.e. $\sigma = G \frac{\partial S}{\partial x}$. Cette dernière équation est exactement la loi de comportement d'un ressort.

Modèle de Maxwell On peut maintenant généraliser le modèle en supposant que :

- $\sigma(\mathbf{x}, t)$ représente le tenseur des contraintes qui est symétrique (théorème 9),
- $\mathbf{u}(\mathbf{x}, t)$ est le champ de vitesse du fluide et satisfait $\nabla \cdot \mathbf{u} = 0$ (fluide incompressible, théorème 4),
- les dérivées temporelles doivent être remplacées par des dérivées dite objectives qui généralisent la notion de dérivée matérielle aux champs de tenseurs. Son expression est fournie par la dérivée temporelle ordinaire mais évaluée dans un repère convecté par le fluide. Dans ce travail, nous utiliserons la dérivée contravariante définie par

$$\frac{\delta \sigma}{\delta t} = \frac{\partial \sigma}{\partial t} + (\mathbf{u} \cdot \nabla) \sigma - \nabla \mathbf{u} \sigma - \sigma (\nabla \mathbf{u})^t$$

- $\frac{\partial \gamma}{\partial t}$ devient le tenseur taux de déformation $2\dot{\gamma}(\mathbf{u})$ où $(\dot{\gamma}(\mathbf{u}))_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right)$.

En généralisant (5.12) et (5.14) nous pouvons énoncer la loi constitutive

$$\begin{cases} \boldsymbol{\sigma} = -p\mathbf{I} + \boldsymbol{\tau} \\ \lambda \frac{\delta \boldsymbol{\tau}}{\delta t} + \boldsymbol{\tau} = 2\nu \dot{\boldsymbol{\gamma}}(\mathbf{u}) \end{cases} \quad (5.15)$$

où $\boldsymbol{\tau}$ représente le tenseur des contraintes supplémentaires.

Équations du problème global Nous pouvons donc formuler les équations différentielles qui gouvernent le comportement d'un fluide viscoélastique stationnaire incompressible en combinant les équations de Stokes (5.11) et les équations constitutives de Maxwell (5.15) :

$$\begin{cases} -\nabla \cdot \boldsymbol{\tau} + \nabla p = \mathbf{f} \\ \nabla \cdot \mathbf{u} = 0 \\ \lambda \frac{\delta \boldsymbol{\tau}}{\delta t} + \boldsymbol{\tau} = 2\nu \dot{\boldsymbol{\gamma}}(\mathbf{u}) \end{cases} \quad (5.16)$$

où $\frac{\delta \boldsymbol{\sigma}}{\delta t} = (\mathbf{u} \cdot \nabla) \boldsymbol{\sigma} - \nabla \mathbf{u} \boldsymbol{\sigma} - \boldsymbol{\sigma} (\nabla \mathbf{u})^t$.

5.4 Discrétisation des équations

Nous allons maintenant discuter de la résolution numérique des systèmes d'équations différentielles dont nous avons discuté aux sections 5.2 et 5.3. Tout d'abord, nous aborderons la résolution du système de Stokes (5.11) qui est un problème classique en éléments finis. Ensuite, nous élaborerons quelques approches possibles pour résoudre le système de Maxwell (5.15). Mentionnons également que les idées et les notations qui ont été introduites au chapitre 1 concernant la résolution du problème de Poisson (1.1), seront réutilisées avec ces systèmes plus complexes.

5.4.1 Résolution numérique du problème de Stokes

Commençons d'abord par préciser le domaine de calcul des équations de Stokes. Posons Ω , un domaine régulier, fermé et connexe de \mathbb{R}^2 dont la frontière est notée Γ . Sur une partie

Γ_0 de cette frontière, nous allons appliquer des conditions aux limites de type Dirichlet tandis que sur $\Gamma \setminus \Gamma_0$ on imposera des conditions de type Neumann homogène.

Avec ces notations, le problème de Stokes s'énonce comme suit :

Trouver \mathbf{u} et p tel que :

$$\left\{ \begin{array}{ll} -2\nu \nabla \cdot \dot{\gamma}(\mathbf{u}) + \nabla p = \mathbf{f} & \text{dans } \Omega, \\ \nabla \cdot \mathbf{u} = 0 & \text{dans } \Omega, \\ \mathbf{u} = \mathbf{u}_0 & \text{sur } \Gamma_0, \\ \nabla \mathbf{u} \cdot \mathbf{n} = 0 & \text{sur } \Gamma \setminus \Gamma_0. \end{array} \right. \quad (5.17)$$

Fixons le cadre fonctionnel afin d'obtenir la formulation variationnelle du problème de Stokes. Nous choisissons comme espaces

$$\mathcal{V} := (H^1(\Omega))^2 = \{v = (v_1, v_2) \in (L^2(\Omega))^2 \mid \nabla v_i \in (L^2(\Omega))^2 \quad i = 1, 2\}, \quad (5.18)$$

$$\mathcal{V}_0 := \{v \in \mathcal{V} \mid v = 0 \text{ p.p. sur } \Gamma_0\} \quad (5.19)$$

et

$$\mathcal{Q} := L^2(\Omega). \quad (5.20)$$

Pour plus de détails sur ces espaces, on peut consulter [16].

Formulation variationnelle Pour obtenir la formulation variationnelle, nous procédons comme nous l'avons fait pour le problème de Poisson (Sect. 1.1). Nous multiplions par des fonctions tests $\mathbf{v} \in \mathcal{V}_0$ et $q \in \mathcal{Q}$. Ainsi, le problème variationnel de Stokes s'écrit :

Trouver $(\mathbf{u}, p) \in \mathcal{V} \times \mathcal{Q}$ tel que $\mathbf{u} - \mathbf{u}_0 \in \mathcal{V}_0$ et

$$\left\{ \begin{array}{ll} 2\nu \int_{\Omega} \dot{\gamma}(\mathbf{u}) : \dot{\gamma}(\mathbf{v}) \, d\mathbf{x} - \int_{\Omega} p \nabla \cdot \mathbf{v} \, d\mathbf{x} = \int_{\Omega} \mathbf{f} \cdot \mathbf{v} \, d\mathbf{x} & \forall \mathbf{v} \in \mathcal{V}_0, \\ \int_{\Omega} q \nabla \cdot \mathbf{u} \, d\mathbf{x} = 0 & \forall q \in \mathcal{Q} \end{array} \right. \quad (5.21)$$

où $\mathbf{A} : \mathbf{B} = \sum_{i,j} A_{ij} B_{ij}$ est la contraction tensorielle de \mathbf{A} et \mathbf{B} .

Par ailleurs, si on définit les formes bilinéaires suivantes

$$a(\mathbf{u}, \mathbf{v}) = \int_{\Omega} 2\nu \dot{\gamma}(\mathbf{u}) : \dot{\gamma}(\mathbf{v}) \, d\mathbf{x} \quad (5.22)$$

$$b(\mathbf{v}, q) = - \int_{\Omega} q \nabla \cdot \mathbf{v} \, dx, \quad (5.23)$$

ainsi que la forme

$$(\mathbf{f}, \mathbf{v}) = \int_{\Omega} \mathbf{f} \cdot \mathbf{v} \, dx, \quad (5.24)$$

on peut réécrire (5.21) sous la forme d'un problème mixte

Trouver $(\mathbf{u}, p) \in \mathcal{V} \times \mathcal{Q}$ tel que $\mathbf{u} - \mathbf{u}_0 \in \mathcal{V}_0$ et

$$\begin{cases} a(\mathbf{u}, \mathbf{v}) + b(\mathbf{v}, p) = (\mathbf{f}, \mathbf{v}) & \forall \mathbf{v} \in \mathcal{V}_0, \\ b(\mathbf{u}, q) = 0 & \forall q \in \mathcal{Q}. \end{cases} \quad (5.25)$$

Il est bien connu que ce problème ainsi posé admet une solution unique dans $\mathcal{V} \times \mathcal{Q}$.

Discrétisation de la formulation variationnelle On se donne deux sous-espaces $\mathcal{V}_h \subset \mathcal{V}$ et $\mathcal{Q}_h \subset \mathcal{Q}$ de dimension finie. On posera $\mathcal{V}_{0h} = \mathcal{V}_h \cap \mathcal{V}_0$.

Le problème discret s'écrit :

Trouver $(\mathbf{u}_h, p_h) \in \mathcal{V}_h \times \mathcal{Q}_h$ tel que $\mathbf{u}_h - \mathbf{u}_{h0} \in \mathcal{V}_{0h}$ et vérifiant

$$\begin{cases} a(\mathbf{u}_h, \mathbf{v}_h) + b(\mathbf{v}_h, p_h) = (\mathbf{f}, \mathbf{v}_h) & \forall \mathbf{v}_h \in \mathcal{V}_{0h}, \\ b(\mathbf{u}_h, q_h) = 0 & \forall q_h \in \mathcal{Q}_h. \end{cases} \quad (5.26)$$

Ce problème discret admet toujours une solution mais ne conduit pas nécessairement à une bonne approximation du problème original. Afin d'obtenir des estimations d'erreur indépendante du maillage, il faut que le problème mixte discret satisfasse une condition de stabilité connue sous le nom de **condition de Brezzi-Babuska** ou encore condition inf-sup. Cette condition est donnée par :

$$\exists \beta \text{ tel que } \inf_{q_h \in \mathcal{Q}_h \setminus \{0\}} \sup_{\mathbf{v}_h \in \mathcal{V}_h \setminus \{0\}} \frac{b(\mathbf{v}_h, q_h)}{\|\mathbf{v}_h\|_{\mathcal{V}} \|q_h\|_{\mathcal{Q}}} \geq \beta > 0.$$

Pour satisfaire la condition inf-sup, il s'agit de choisir convenablement la paire d'éléments finis $(\mathcal{V}_h, \mathcal{Q}_h)$ utilisés. De tels éléments finis sont dit stables et il en existe plusieurs types. Voir [4]. Pour les tests numérique, nous choisirons un élément fini continu stable très connu qui porte le nom de Taylor-Hood. Cet élément correspond au choix des sous-espaces d'éléments finis suivants :

$$\mathcal{V}_h = \{ \mathbf{v}_h \in (C^0(\bar{\Omega}))^2 \mid \mathbf{v}_h|_K \in (P_2(K))^2 \ \forall K \in \mathcal{T} \}$$

$$\mathcal{Q}_h = \{ q_h \in (C^0(\bar{\Omega}))^2 \mid q_h|_K \in (P_1(K))^2 \ \forall K \in \mathcal{T} \}$$

où $P_k(K)$ est l'espace des polynômes de degré au plus k définis sur un élément K d'une triangulation \mathcal{T} . Cet élément est souvent appelé l'élément $P_2 - P_1$. Pour plus de renseignements sur la stabilité de cet élément consulter [4].

Passons à la forme matricielle associée au système discret. Pour tenir compte des conditions aux limites de type Dirichlet, nous préférons résoudre le système (5.26) en correction. Ainsi, nous sommes conduit à un système linéaire par blocs de la forme

$$\begin{pmatrix} \mathbf{A} & \mathbf{B}^t \\ \mathbf{B} & 0 \end{pmatrix} \begin{pmatrix} \delta \mathbf{u}_h \\ \delta p_h \end{pmatrix} = \begin{pmatrix} R_{\mathbf{u}} \\ R_p \end{pmatrix} \quad (5.27)$$

où les blocs $\mathbf{A} = \{a_{ij}\}$, $\mathbf{B} = \{b_{ij}\}$, $R_{\mathbf{u}}$ et R_p correspondent à la discrétisation des termes :

$$\begin{cases} a_{ij} = \int_{\Omega} 2\nu \dot{\gamma}(\mathbf{v}_h) : \dot{\gamma}(\mathbf{w}_h) \, d\mathbf{x} & \mathbf{v}_h, \mathbf{w}_h \in \mathcal{V}_{0h} \\ b_{ij} = - \int_{\Omega} q_h \nabla \cdot \mathbf{v}_h \, d\mathbf{x} & \mathbf{v}_h \in \mathcal{V}_{0h}, q_h \in \mathcal{Q}_h \\ F_i = \int_{\Omega} \mathbf{f} \cdot \mathbf{v}_h \, d\mathbf{x} & \mathbf{v}_h \in \mathcal{V}_{0h} \\ R_{\mathbf{u}} = F - \mathbf{A} \mathbf{u}_{0h} \\ R_p = -\mathbf{B} \mathbf{u}_{0h} \end{cases}$$

Nous pouvons remarquer que le système linéaire (5.27) comporte des éléments nuls sur la diagonale. De plus, s'il n'y a pas de conditions aux limites de type Neumann, alors le problème admet une infinité de solutions. Cela est dû au fait que la pression est définie à une constante près. Pour remédier à cette difficulté, nous pouvons fixer la pression à un sommet, mais cette pratique est plus difficile à implanter. Nous allons plutôt ajouter

un coefficient de pénalisation $\epsilon > 0$ petit. Pour ce faire, nous introduisons un terme de pénalisation dans la deuxième équation (5.21) qui devient

$$\int_{\Omega} q_h \nabla \cdot \mathbf{u}_h \, d\mathbf{x} + \int_{\Omega} \epsilon p_h q_h \, d\mathbf{x} = 0 \quad \forall q_h \in \mathcal{Q}_h.$$

De cette manière, le système admet une seule solution et s'écrit

$$\begin{pmatrix} \mathbf{A} & \mathbf{B}^t \\ \mathbf{B} & -\epsilon \mathbf{M} \end{pmatrix} \begin{pmatrix} \delta \mathbf{u}_h \\ \delta p_h \end{pmatrix} = \begin{pmatrix} R_u \\ R_p \end{pmatrix} \quad (5.28)$$

où \mathbf{M} est la matrice masse définie par $\{m_{ij}\} = \int_{\Omega} p_h q_h \, d\mathbf{x}$ avec $p_h, q_h \in \mathcal{Q}_h$. Notons que l'ajout de ce terme perturbe les solutions du problème variationnel discret de l'ordre du terme ϵ . Voir [3] pour plus de détails.

5.4.2 Problème global

Rappelons que le système couplé (5.16) ayant comme inconnues les trois champs $(\mathbf{u}, p, \boldsymbol{\tau})$ s'écrit comme un problème différentiel :

Trouver $(\mathbf{u}, p$ et $\boldsymbol{\tau})$ tel que

$$\begin{cases} -\nabla \cdot \boldsymbol{\tau} + \nabla p = \mathbf{f}, \\ \nabla \cdot \mathbf{u} = 0, \\ \lambda \frac{\delta \boldsymbol{\tau}}{\delta t} + \boldsymbol{\tau} = 2\nu \dot{\boldsymbol{\gamma}}(\mathbf{u}). \end{cases} \quad (5.29)$$

À ce système s'ajoute les conditions aux limites du problème en vitesse tel que vu à la section précédente ainsi qu'une condition aux limites pour la variable $\boldsymbol{\tau}$. Étant donné que l'équation constitutive est de type hyperbolique (pour \mathbf{u} fixé), il faut imposer la valeur de $\boldsymbol{\tau}$ uniquement à la partie entrante Γ_e du fluide, c'est-à-dire

$$\Gamma_e = \{\mathbf{x} \in \Gamma \mid \mathbf{u}(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}) < 0\}.$$

Pour obtenir la formulation variationnelle de ce problème, nous procédons de la même manière qu'à la section précédente. Nous choisirons l'espace

$$\Sigma := \{\boldsymbol{\tau} \in (L^2(\Omega))^3 \mid \tau_{ij} \in (L^2(\Omega))^3 \quad i \leq j = 1, 2\}$$

et nous noterons par Σ_0 l'espace

$$\Sigma_0 := \{\boldsymbol{\tau} \in \Sigma \mid \boldsymbol{\tau} = \boldsymbol{\tau}_0 \text{ p.p. sur } \Gamma_e\}.$$

En multipliant par $(\mathbf{v}, q, \phi) \in \mathcal{V}_0 \times \mathcal{Q} \times \Sigma$, ceci nous permet d'écrire la formulation variationnelle du problème couplé :

Trouver $(\mathbf{u}, p, \boldsymbol{\tau}) \in \mathcal{V} \times \mathcal{Q} \times \Sigma$ tel que $\mathbf{u} - \mathbf{u}_0 \in \mathcal{V}_0$, $\boldsymbol{\tau} - \boldsymbol{\tau}_0 \in \Sigma_0$ et vérifiant $\forall (\mathbf{v}, q, \phi) \in \mathcal{V}_0 \times \mathcal{Q} \times \Sigma$

$$\begin{cases} \int_{\Omega} \boldsymbol{\tau} : \dot{\boldsymbol{\gamma}}(\mathbf{v}) \, d\mathbf{x} - \int_{\Omega} p \nabla \cdot \mathbf{v} \, d\mathbf{x} = \int_{\Omega} \mathbf{f} \cdot \mathbf{v} \, d\mathbf{x} \\ \int_{\Omega} q \nabla \cdot \mathbf{u} \, d\mathbf{x} = 0 \\ \int_{\Omega} \{\boldsymbol{\tau} + \lambda \frac{d\boldsymbol{\tau}}{dt} - 2\nu \dot{\boldsymbol{\gamma}}(\mathbf{u})\} : \phi \, d\mathbf{x} = 0. \end{cases} \quad (5.30)$$

La discrétisation de ce système pose de nombreux problèmes même dans le cas limite où $\lambda = 0$. Nous suggérons au lecteur de consulter [8] pour obtenir de plus amples informations sur les espaces de discrétisation permettant d'approcher ce problème couplé.

Étant donné notre désir de tester les capacités de l'adaptation de maillages, nous avons adopté une approche simplifiée de type découplée qui est souvent utilisée par les ingénieurs. Elle consiste à résoudre séparément les variables (\mathbf{u}, p) et $\boldsymbol{\sigma}$. De manière plus précise, au lieu de résoudre les deux premières équations du système (5.30), on résout l'équation de Stokes avec η comme viscosité du fluide newtonien. Cette solution ne tient pas compte du tenseur des contraintes supplémentaires du fluide viscoélastique. Ensuite, on reporte ce champ de vitesse dans la loi de comportement de Maxwell qui constitue une équation de type transport. Cela nous permettra de créer des maillages fins ayant un nombre élevé de sommets et d'éléments. Cette méthode est avantageuse pour nous puisqu'il est très coûteux de résoudre le problème variationnel couplé (5.30). Les sections suivantes décrivent la discrétisation de l'équation de comportement de Maxwell.

5.4.3 Les méthodes de décentrage selon les lignes de courant

La loi de comportement constitue une équation de type hyperbolique. Or il est bien

connu que l'utilisation directe d'une discrétisation de type Galerkin conduit à des instabilités numériques.

Dans cette section, nous allons discuter de deux méthodes utilisées pour contrer ces problèmes d'instabilité dû à la présence du terme de transport $(\mathbf{u} \cdot \nabla)\boldsymbol{\tau}$ apparaissant dans l'expression de la dérivée convective $\frac{\delta \boldsymbol{\tau}}{\delta t}$. De plus, comme certains des problèmes étudiés contiennent des singularités, les schémas ont tendance à engendrer des oscillations de la solution et ce, dans les régions qui entourent les singularités. Ces oscillations sont plus nombreuses lorsque le coefficient d'élasticité λ augmente. C'est pourquoi nous devons avoir recours à une stabilisation du schéma par une méthode de décentrage. Nous aborderons deux méthodes bien connues, à savoir, la méthode Streamline-Upwind/Petrov-Galerkin (SUPG) et la méthode Streamline-Upwind (SU). Ces méthodes ont pour but d'ajouter un terme de diffusion qui stabilisera le problème discret.

Méthode Streamline-Upwind/Petrov-Galerkin (SUPG) Soit Σ_h un sous-espace de dimension finie approximant Σ . Cette méthode consiste à modifier les fonctions tests $\phi_h \in \Sigma_h$ en leurs ajoutant un terme de diffusion qui agit seulement dans le sens des lignes de courant. Les fonctions tests ainsi perturbées sont données par

$$\bar{\phi}_h = \phi_h + \kappa(\mathbf{w}_h \cdot \nabla)\phi_h \quad (5.31)$$

où $\mathbf{w}_h = \frac{\mathbf{u}_h}{\|\mathbf{u}_h\|}$ est la direction locale du courant et κ une pondération du terme de diffusion par la taille des triangles. En remplaçant ϕ_h par $\bar{\phi}_h$ dans l'équation de Maxwell de (5.30) nous obtenons la nouvelle formulation variationnelle suivante :

$$\int_{\Omega} \left\{ \boldsymbol{\tau}_h + \lambda \frac{\delta \boldsymbol{\tau}_h}{\delta t} - 2\nu \dot{\gamma}(\mathbf{u}_h) \right\} : \{ \phi_h + \kappa(\mathbf{w}_h \cdot \nabla)\phi_h \} d\mathbf{x} = 0 \quad \forall \phi_h \in \Sigma_h$$

Cette méthode est dite consistante car les solutions du problème continu vérifient la version continue de cette équation.

Méthode Streamline-Upwind(SU) La méthode SU utilise la même perturbation que celle de SUPG, i.e. (5.31). Cependant on n'utilise cette modification de la fonction test

que pour le terme de transport $(\mathbf{u} \cdot \nabla)\boldsymbol{\tau}$. La nouvelle formulation variationnelle s'écrit ainsi

$$\int_{\Omega} \left\{ \boldsymbol{\tau}_h + \lambda \frac{\delta \boldsymbol{\tau}_h}{\delta t} - 2\nu \dot{\boldsymbol{\gamma}}(\mathbf{u}_h) \right\} : \boldsymbol{\phi}_h \, d\mathbf{x} + \int_{\Omega} \lambda (\mathbf{u}_h \cdot \nabla) \boldsymbol{\tau}_h : \kappa(\mathbf{w}_h \cdot \nabla) \boldsymbol{\phi}_h \, d\mathbf{x} = 0 \quad \forall \boldsymbol{\phi}_h \in \Sigma_h.$$

Notons que l'ajout du terme de diffusion perturbe le problème variationnel et ce qui fait qu'on ne résout plus exactement le problème original. Ceci explique pourquoi cette méthode est souvent appelée inconsistante.

5.4.4 Discrétisation de la loi de comportement de Maxwell

Dans les tests numériques de la prochaine section, nous avons choisi le schéma SU comme méthode de décentrage de l'équation de la loi de comportement de Maxwell. Ce choix se base sur le fait que cette méthode stabilise mieux que SUPG pour la classe de problèmes étudiés ayant des singularités au niveau des contraintes.

De plus nous choisissons un élément fini de type P_1 pour approcher les différentes composantes de la variable $\boldsymbol{\tau}_h$. Plus précisément, l'espace Σ_h sera définie par

$$\Sigma_h := \{ \boldsymbol{\tau}_h \in (C^0(\bar{\Omega}))^3 \mid \boldsymbol{\tau}_{hK} \in (P_1(K))^3 \quad \forall K \in \mathcal{T} \}$$

et

$$\Sigma_{0h} := \Sigma_h \cap \Sigma_0.$$

Alors nous obtenons la nouvelle formulation variationnelle discrète pour l'équation constitutive de Maxwell :

Trouver $\boldsymbol{\tau}_h \in \Sigma_h$ tel que $\boldsymbol{\tau}_h - \boldsymbol{\tau}_{0h} \in \Sigma_{0h}$ vérifiant

$$\int_{\Omega} \left\{ \boldsymbol{\tau}_h + \lambda \frac{\delta \boldsymbol{\tau}_h}{\delta t} - 2\nu \dot{\boldsymbol{\gamma}}(\mathbf{u}_h) \right\} : \boldsymbol{\phi}_h \, d\mathbf{x} + \int_{\Omega} \lambda (\mathbf{u}_h \cdot \nabla) \boldsymbol{\tau}_h : \kappa_h(\mathbf{w}_h \cdot \nabla) \boldsymbol{\phi}_h \, d\mathbf{x} = 0 \quad \forall \boldsymbol{\phi}_h \in \Sigma_{0h}. \quad (5.32)$$

5.5 Résultats numériques

Dans ce chapitre, nous appliquerons la méthode d'adaptation de maillages sur quatre problèmes types. Les deux premiers sont le problème de Poiseuille et le problème de la couche limite qui ont tous deux une solution exacte pour la loi de comportement de Maxwell. Les deux autres problèmes, qui sont la contraction et le problème de l'écoulement autour d'un cylindre, n'ont pas de solution exacte. Nous testerons tous ces problèmes dans le cas newtonien ($\lambda = 0.0$) et dans le cas viscoélastique ($\lambda = 1.0$) suivant une approche simplifiée. Rappelons que cette approche consiste à résoudre le problème de Stokes pour les variables (\mathbf{u}_h, p_h) et par la suite, l'équation constitutive de Maxwell pour trouver la variable discrète $\boldsymbol{\tau}_h$ en utilisant comme champ de transport le champ de vitesse fourni précédemment. Cette dernière nous permettra de construire notre estimateur *a posteriori* afin de calculer la métrique telle que décrite dans la section 2.2.2. Pour cela, nous définissons la variable S par la contraction matricielle $\sqrt{\boldsymbol{\tau} : \boldsymbol{\tau}}$ i.e. $S = \sqrt{\tau_{11}^2 + 2\tau_{12}^2 + \tau_{22}^2}$, car $\boldsymbol{\tau}$ est symétrique. Pour chaque problème, nous effectuerons deux tests en posant $\lambda = 0.0$ et $\lambda = 1.0$. À noter que pour l'ensemble des tests, la viscosité fut fixée à $\nu = 1.0$.

Pour chaque problème, les caractéristiques reliées aux maillages résultant de l'adaptation seront mises sous forme de tableaux. De plus, pour les problèmes qui ont une solution analytique, nous présenterons dans un tableau, les erreurs commises entre la solution analytique évaluée aux sommets du maillage et la solution calculée. Ces erreurs seront évaluées par la norme $\|\cdot\|_\infty$ et $\|\cdot\|_2$ de la différence entre les deux solutions. De plus, pour chaque maillage obtenu, nous ajouterons la figure correspondante. Nous inclurons aussi plusieurs graphiques pertinents reliés aux variables discrètes.

Pour chacun de ces problèmes, nous avons fixé une longueur désirée l_d de manière à utiliser un ordinateur personnel pour résoudre les problèmes EF. L'ordinateur utilisé possède 128 MégaOctets de mémoire vive et 500 MégaOctets de mémoire disque. Nous allons indiquer, pour chaque problème, l'espace mémoire utilisé pour stocker la matrice compressée (i.e. les entrées non nulles). Puisque nous utilisons une méthode de résolution

directe (LU) pour résoudre le système matriciel, ce chiffre ne donne pas une image réelle de la mémoire utilisée. En effet, la méthode LU exige une structure de donnée particulière qui nécessite un important remplissage de la matrice creuse. Dans la pratique, nous observons que la mémoire réelle utilisée par la méthode LU est au moins 10 fois l'espace mémoire nécessaire pour stocker la matrice.

5.5.1 Écoulement de Poiseuille

Nous aborderons d'abord le problème de l'écoulement de Poiseuille qui est un problème classique de la mécanique des fluides. Nous nous intéressons à ce problème puisque la solution exacte est connue et que de nombreux tests ont démontré expérimentalement la validité de la solution analytique dans le cas de fluide newtonien à basse vitesse (Voir [6]). Cet écoulement est décrit par un fluide qui se déplace sous l'influence d'un gradient de pression dans un canal situé entre deux plaques immobiles. La vitesse de ce fluide est uniquement dans le sens de l'axe des x avec un profil parabolique. Nous pouvons voir à la figure 5.2 la géométrie du canal qui est un rectangle de 1×5 dont toutes les arêtes sont droites. Nous supposons aussi la condition d'adhérence du fluide à la paroi. Cette condition implique que la vitesse du fluide est nulle sur les parois du canal. Enfin, nous supposons qu'il n'y a pas de force externe qui s'applique sur le système (i.e. $f = (0, 0)$ dans (5.11)). Pour résoudre ce problème, nous devons imposer les conditions aux limites

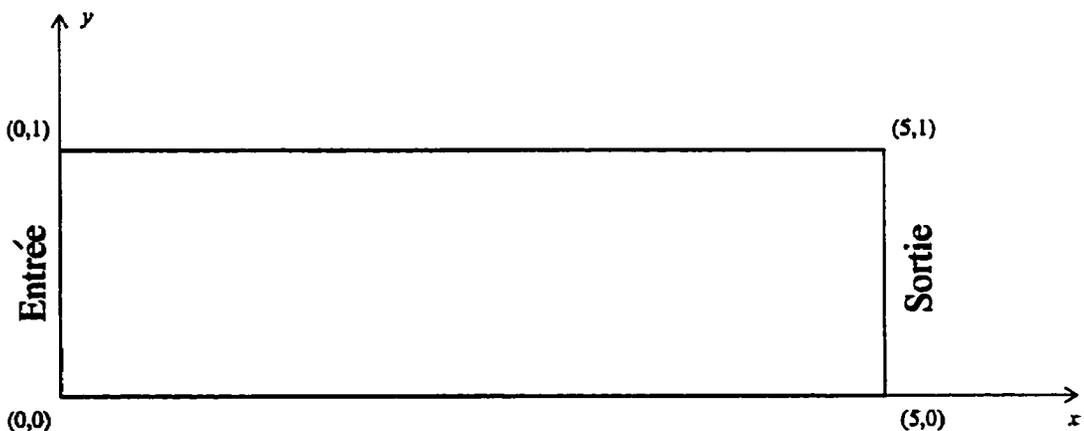


FIG. 5.2 – Géométrie du canal.

de type Dirichlet qui suivent.

D'abord, lors de la résolution du problème de Stokes, nous imposons la solution exacte donnée par l'équation (5.33) à la variable \mathbf{u} et ce, à l'entrée et à la sortie.

$$\begin{cases} \mathbf{u} = (u_1, u_2) = (1 - (2y - 1)^2, 0) \\ p = -8\nu x + \text{Constante}. \end{cases} \quad (5.33)$$

Cette solution sera utilisée pour valider les résultats obtenus. Par ailleurs, comme nous avons supposé que le fluide adhère à la surface du canal, nous devons imposer que le fluide ait une vitesse nulle $\mathbf{u} = (0, 0)$ sur les parois du canal. Enfin, nous imposons que $f = (0, 0)$ puisque nous avons supposé qu'aucune force externe ne s'applique au système.

Pour la partie viscoélastique (problème de Maxwell), nous imposons seulement à l'entrée la solution exacte pour $\boldsymbol{\tau}$ donnée par l'équation (5.34)

$$\begin{cases} \tau_{11} = 2\lambda\nu \left(\frac{\partial u_1}{\partial y}\right)^2 = 32\lambda\nu (2y - 1)^2 \\ \tau_{12} = \nu \frac{\partial u_1}{\partial y} = -4\nu (2y - 1) \\ \tau_{22} = 0 \end{cases} \quad (5.34)$$

Lors de la résolution du problème de Maxwell, nous utiliserons la solution du problème de Stokes pour initialiser la variable \mathbf{u}_h qui apparaît dans la formulation variationnelle discrète (5.32).

Les deux tests que nous effectuerons seront détaillés à la section 5.5.2 pour le cas newtonien ($\lambda = 0.0$) et à la section 5.5.3 pour le cas viscoélastique ($\lambda = 1.0$).

5.5.2 Résolution du problème de Poiseuille : $\lambda = 0.0$

Pour ce cas, nous avons effectué trois séquences d'adaptation pour vérifier la convergence de la méthode d'adaptation. La longueur désirée fut fixée à $l_d = 1.0$. Les autres données relatives aux résultats d'adaptation sont indiquées au tableaux 5.2 et 5.3. Le tableau 5.1

indique l'erreur commise entre la solution calculée et la solution analytique qui est donnée par (5.33) et (5.34) et qui est évaluée aux sommets du maillages. Nous pouvons d'abord remarquer que les erreurs sont de l'ordre de 10^{-6} . Comme la solution exacte se trouve dans l'espace de discrétisation, nous espérons trouver une erreur plus petite. Mais la pénalisation introduit une erreur de l'ordre du coefficient de pénalisation et c'est bien ce que nous observons (voir [3]). En outre, la méthode SU introduit, elle aussi, une erreur due au schéma inconsistant. Somme toute, les solutions sont très satisfaisantes. Pour ce problème, l'adaptation n'a pas contribué à faire diminuer l'erreur qui est presque nulle. Il est bien de noter que certains estimateurs hiérarchiques, basés sur l'erreur d'interpolation, peuvent conduire à un maillage très déraffiné puisque la solution discrète est exacte pour n'importe quels maillages. Nous ne pouvons nous attendre à un même comportement ici dû à la nature de notre estimateur *a posteriori*.

Nous pouvons voir le maillage original (nommé ORIGINAL) utilisé pour le problème de Poiseuille à la figure 5.3 et les maillages adaptés ADAPT1, ADAPT2 et ADAPT3 correspondant aux différentes adaptations successives à la figure 5.4. Les caractéristiques de ces maillages sont décrites aux tableaux 5.2 et 5.3. Nous y voyons que la qualité du maillage de départ est plus élevée que celles des autres maillages. C'est souvent le cas pour les maillages qui ont été générés de façon à obtenir un maillage uniforme et régulier. Les autres maillages présentent une qualité moyenne très acceptable. Au tableau 5.3, nous pouvons voir entre autre, le nombre de raffinements et de déraffinements effectués pour obtenir les maillages adaptés.

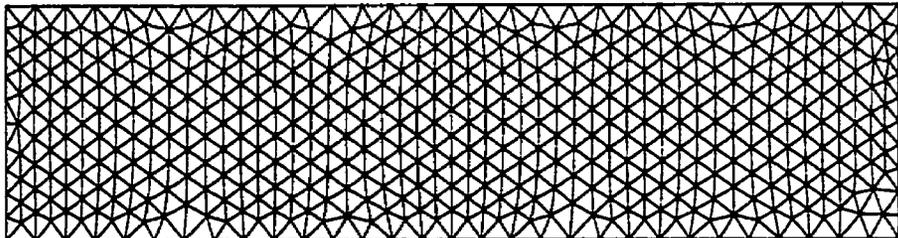


FIG. 5.3 – Maillage ORIGINAL du canal.

	Maillages			
Erreur	ORIGINAL	ADAPT1	ADAPT2	ADAPT3
$\ (\mathbf{u}_h)_x - (\pi_h \mathbf{u})_x\ _2$	1.78838e-06	1.78838e-06	1.78838e-06	1.78838e-06
$\ (\mathbf{u}_h)_x - (\pi_h \mathbf{u})_x\ _\infty$	1.499e-06	1.499e-06	1.499e-06	1.499e-06
$\ (\mathbf{u}_h)_y - (\pi_h \mathbf{u})_y\ _2$	3.58513e-08	3.58645e-08	3.58644e-08	3.58775e-08
$\ (\mathbf{u}_h)_y - (\pi_h \mathbf{u})_y\ _\infty$	4.19349e-08	4.2179e-08	4.21493e-08	4.21168e-08
$\ p_h - \pi_h p\ _2$	6.35877e-05	6.35878e-05	6.35879e-05	6.35881e-05
$\ p_h - \pi_h p\ _\infty$	4.31084e-05	4.34139e-05	4.33366e-05	4.3133e-05
$\ (\boldsymbol{\tau}_h)_{11} - (\pi_h \boldsymbol{\tau})_{11}\ _2$	1.12641e-06	1.12149e-06	1.12119e-06	1.11762e-06
$\ (\boldsymbol{\tau}_h)_{11} - (\pi_h \boldsymbol{\tau})_{11}\ _\infty$	1.28374e-06	1.61483e-06	1.44507e-06	1.6171e-06
$\ (\boldsymbol{\tau}_h)_{12} - (\pi_h \boldsymbol{\tau})_{12}\ _2$	5.64778e-06	5.6477e-06	5.6477e-06	5.64766e-06
$\ (\boldsymbol{\tau}_h)_{12} - (\pi_h \boldsymbol{\tau})_{12}\ _\infty$	6.01611e-06	6.01616e-06	6.01631e-06	6.01651e-06
$\ (\boldsymbol{\tau}_h)_{22} - (\pi_h \boldsymbol{\tau})_{22}\ _2$	4.63865e-07	4.64438e-07	4.64263e-07	4.63869e-07
$\ (\boldsymbol{\tau}_h)_{22} - (\pi_h \boldsymbol{\tau})_{22}\ _\infty$	9.33747e-07	9.98129e-07	9.77139e-07	9.8641e-07
Nombre d'éléments	938	1084	1062	1007
Nombre de sommets	510	605	602	578

TAB. 5.1 – Résultats du problème de Poiseuille : Cas $\lambda = 0.0$.

	Maillages			
Qualité	ORIGINAL	ADAPT1	ADAPT2	ADAPT3
Q_K moyenne	0.968861	0.837344	0.854922	0.845747
Q_K minimum	0.708772	0.226448	0.219721	0.209225
Q_K maximum	0.999993	0.999988	0.999539	0.999766
Nombre d'éléments	938	1084	1062	1007
Nombre de sommets	510	605	602	578

TAB. 5.2 – Caractéristiques des maillages générés pour le problème de Poiseuille : Cas $\lambda = 0.0$.

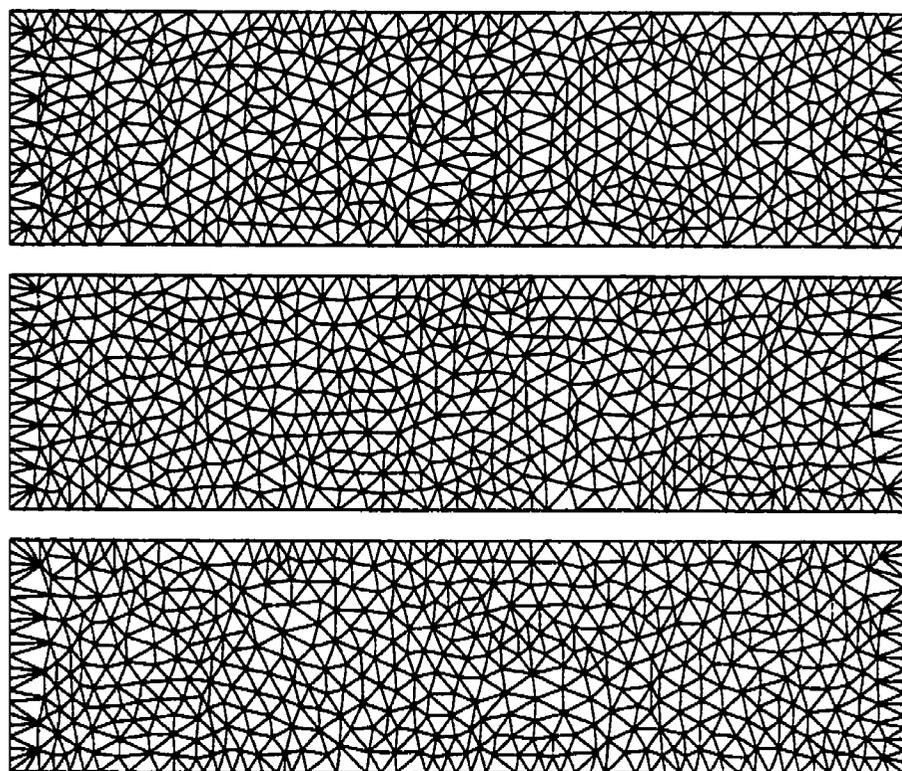


FIG. 5.4 – Maillages ADAPT1, ADAPT2 et ADAPT3 (de haut en bas) générés pour le problème de Poiseuille : Cas $\lambda = 0.0$.

	Maillages		
	ADAPT1	ADAPT2	ADAPT3
Longueur calculée l_c	1.0	1.0	1.0
Nb raf. d'arêtes	421	351	335
Nb déraf. d'arêtes	350	370	366
Nb ret. d'arêtes	1028	903	886
Nb dép. sommets	6820	6652	6268
Dép. sommet Moyen	0.0037611	0.00214049	0.0215474
Dép. sommet Max.	0.0224844	0.0186854	0.00270782
Temps d'adaptation (sec.)	63.08	60.52	59.5
Mémoire utilisée (Octets)	1 840 488	1 809 816	1721256
Dimension Syst. Lin. (Nb. lignes)	5191	5132	4902

TAB. 5.3 – Données relatives à l'adaptation pour le problème de Poiseuille : Cas $\lambda = 0.0$.

5.5.3 Résolution du problème de Poiseuille : $\lambda = 1.0$

Contrairement au problème newtonien de Poiseuille, ce problème comporte une variable qui ne se trouve pas dans l'espace de discrétisation. La variable τ_{11} est d'ordre 2. Nous nous serions attendus à ce que la méthode d'adaptation raffine le maillage de façon à diminuer progressivement l'erreur sur τ_{11} . Cependant, nous avons eu la surprise de constater le contraire. L'erreur commise augmentait bien que les maillages devenaient plus fins. Nous avons d'abord questionné l'estimateur pour nous rendre compte par la suite que cette erreur était peut-être due à l'orientation des arêtes. Pour en arriver à cette conclusion, nous avons refait une adaptation à partir du maillage de départ (ORIGINAL) mais en ne permettant pas le déraffinement d'arêtes. Le maillage obtenu (SANS DER) est illustré à la figure 5.6 et nous pouvons voir à la figure 5.5, l'erreur $\|(\tau_h)_{11} - (\pi_h \tau)_{11}\|_\infty$. Nous remarquons que cet effet est très prononcé à l'interface entre la zone adaptée et la zone régulière du maillage. Il est intéressant de voir que l'erreur près du bord est très petite.

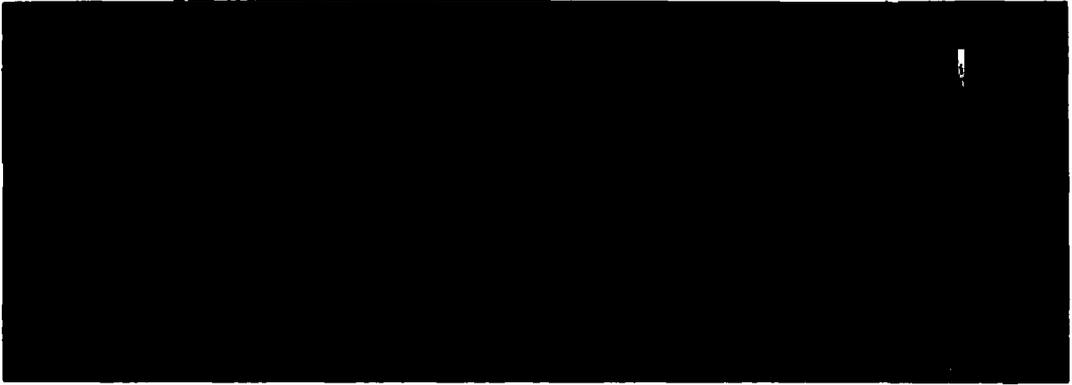


FIG. 5.5 – L'erreur $\|(\tau_h)_{11} - (\pi_h \tau)_{11}\|_\infty$ sur le maillage SANSDER.

Pour les autres variables, la solution exacte se trouve dans l'espace de discrétisation. Nous remarquons toujours une erreur proportionnelle à la pénalisation. La longueur désirée fut fixée à $l_d = 6.0$. Nous pouvons voir le comportement de l'erreur sur ces variables au tableau 5.4.

Il n'en demeure pas moins que la taille de l'erreur sur τ_{11} reste surprenante et que ce phénomène devrait être étudié de façon plus approfondie. La possibilité d'un couplage entre la pénalisation et l'utilisation de la méthode SU serait une avenue à emprunter.

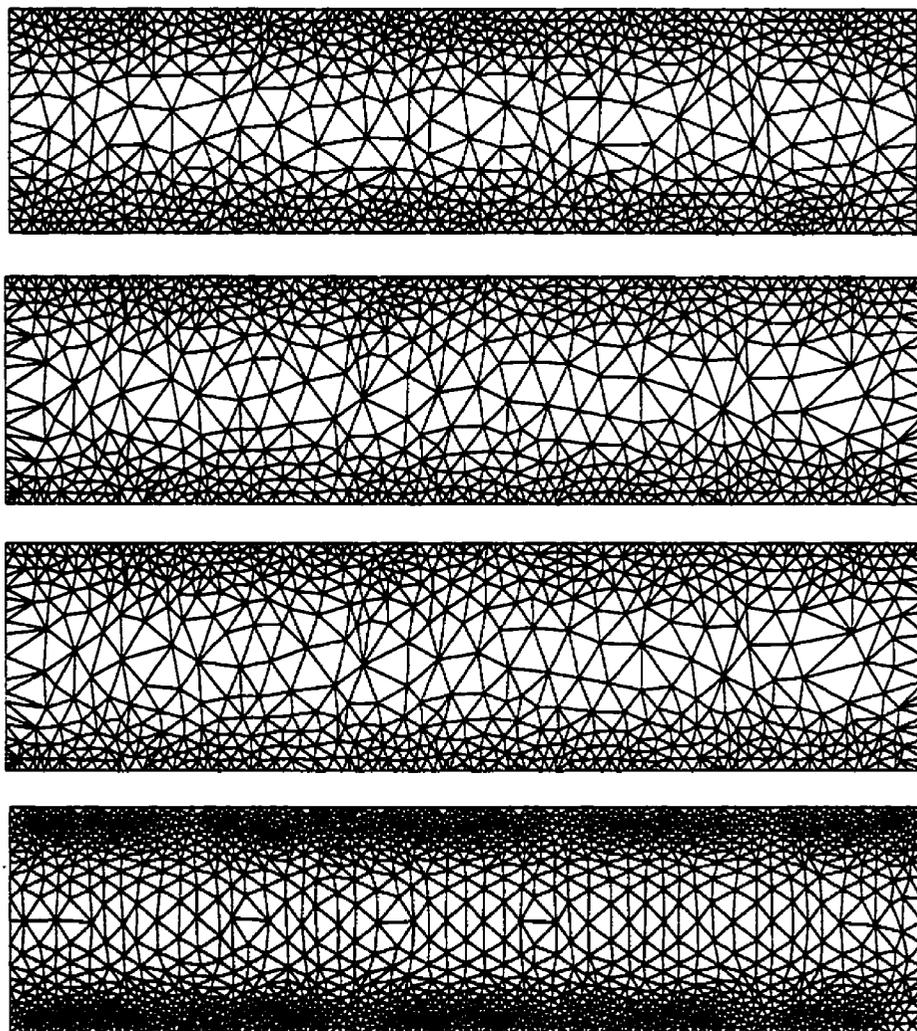


FIG. 5.6 – Maillages ADAPT1, ADAPT2, ADAPT3 et SANSDER (de haut en bas) générés pour le problème de Poiseuille : Cas $\lambda = 1.0$.

	Maillages				
Erreur	ORIGINAL	ADAPT1	ADAPT2	ADAPT3	SANS DER
$\ (\mathbf{u}_h)_x - (\pi_h \mathbf{u})_x\ _2$	1.78838e-06	1.78838e-06	1.78838e-06	1.78838e-06	1.78838e-06
$\ (\mathbf{u}_h)_x - (\pi_h \mathbf{u})_x\ _\infty$	1.499e-06	1.499e-06	1.499e-06	1.49899e-06	1.499e-06
$\ (\mathbf{u}_h)_y - (\pi_h \mathbf{u})_y\ _2$	3.58513e-08	3.5849e-08	3.58553e-08	3.58574e-08	3.58412e-08
$\ (\mathbf{u}_h)_y - (\pi_h \mathbf{u})_y\ _\infty$	4.19349e-08	4.21717e-08	4.2333e-08	4.22685e-08	4.20998e-08
$\ p_h - \pi_h p\ _2$	6.35877e-05	6.35877e-05	6.35879e-05	6.35879e-05	6.35876e-05
$\ p_h - \pi_h p\ _\infty$	4.31084e-05	4.3658e-05	4.40034e-05	4.38916e-05	4.37555e-05
$\ (\boldsymbol{\tau}_h)_{11} - (\pi_h \boldsymbol{\tau})_{11}\ _2$	0.436949	0.701118	0.768357	0.769458	0.749143
$\ (\boldsymbol{\tau}_h)_{11} - (\pi_h \boldsymbol{\tau})_{11}\ _\infty$	0.653338	1.13524	1.10316	1.2375	0.972113
$\ (\boldsymbol{\tau}_h)_{12} - (\pi_h \boldsymbol{\tau})_{12}\ _2$	6.75159e-06	6.76209e-06	6.75239e-06	6.75433e-06	6.78759e-06
$\ (\boldsymbol{\tau}_h)_{12} - (\pi_h \boldsymbol{\tau})_{12}\ _\infty$	7.03835e-06	6.99455e-06	7.0044e-06	6.99812e-06	6.97927e-06
$\ (\boldsymbol{\tau}_h)_{22} - (\pi_h \boldsymbol{\tau})_{22}\ _2$	4.16045e-07	4.1772e-07	4.16054e-07	4.16261e-07	4.23396e-07
$\ (\boldsymbol{\tau}_h)_{22} - (\pi_h \boldsymbol{\tau})_{22}\ _\infty$	7.36237e-07	8.23381e-07	8.31319e-07	8.66385e-07	8.0236e-07
Nombre d'éléments	938	1459	1327	1372	3106
Nombre de sommets	510	828	781	819	1659

TAB. 5.4 – Résultats du problème de Poiseuille : Cas $\lambda = 1.0$.

	Maillages				
Qualité	ORIGINAL	ADAPT1	ADAPT2	ADAPT3	SANS DER
Q_K moyenne	0.968861	0.852735	0.829499	0.821688	0.862484
Q_K minimum	0.708772	0.286614	0.170305	0.1634	0.369212
Q_K maximum	0.999993	0.999957	0.999992	0.99987	0.999981
Nombre d'éléments	938	1459	1327	1372	3106
Nombre de sommets	510	828	781	819	1659

TAB. 5.5 – Caractéristiques des maillages générés pour le problème de Poiseuille : Cas $\lambda = 1.0$.

	Maillages		
	ADAPT1	ADAPT2	ADAPT3
Longueur calculée l_c	6.0	6.0	6.0
Nb raf. d'arêtes	1076	783	867
Nb déraf. d'arêtes	816	863	854
Nb ret. d'arêtes	1559	1716	1930
Nb dép. sommets	10035	9132	9222
Dép. sommet Moyen	0.00212123	0.00276189	0.00244475
Dép. sommet Max.	0.0246372	0.0356701	0.0216156
Temps d'adaptation (sec.)	93.73	90.76	95.84
Mémoire utilisée (Octets)	2 487 048	2 282 136	2 367 816
Dimension Syst. Lin. (Nb. lignes)	7056	6557	6837

TAB. 5.6 – Données relatives à l'adaptation pour le problème de Poiseuille : Cas $\lambda = 1.0$.

5.5.4 Problème de la couche limite

Nous allons maintenant étudier le problème de la couche limite. Ce dernier est fort intéressant parce que nous avons une solution exacte pour tester les résultats d'adaptation. Contrairement au problème de Poiseuille, la majorité des solutions discrètes sont différentes de leurs contreparties exactes. Il devient donc pertinent de comparer l'erreur commise sur chaque maillage. Notons que le problème de la couche limite n'est pas un problème physique réel car nous devons bien choisir les forces externes pour que le fluide ait une solution analytique. L'écoulement a lieu dans le même canal que le problème de Poiseuille (voir la géométrie à la figure 5.2). Le profil d'entrée et de sortie est polynômial d'ordre $2n$. Nous supposons toujours la condition d'adhérence du fluide à la paroi. Nous allons détailler la classe des problèmes de type couche limite, qui sont fonction du paramètre $n > 0$. Nous choisirons par la suite $n = 5$ pour effectuer les calculs.

Ce problème est obtenu en choisissant \mathbf{u} , p et \mathbf{f} de la forme suivante

$$\begin{cases} \mathbf{u} = (u_1, u_2) = (1 - (2y - 1)^{2n}, 0) \\ p = -x^2 \\ \mathbf{f} = (f_1, f_2) = -\nabla \cdot \boldsymbol{\tau} + \nabla p = (\nu 8n(2n - 1)(2y - 1)^{2n-2} - 2x, 0), \end{cases} \quad (5.35)$$

de sorte que le problème de Stokes (5.11) soit vérifié. Par conséquent, lors de la résolution numérique, nous imposons ce profil de vitesse comme condition aux limites à l'entrée et à la sortie.

Par ailleurs, comme nous avons supposé que le fluide adhère à la surface du canal, nous devons imposer que le fluide ait une vitesse nulle $\mathbf{u} = (0, 0)$ sur les parois. Cette condition est vérifiée par (5.35).

Pour la résolution du tenseur des contraintes supplémentaires, nous imposons seulement à l'entrée la solution du problème de Maxwell pour $\boldsymbol{\tau}$ dans le cas d'un écoulement en cisaillement. Cette solution exacte est donnée par l'équation (5.36).

$$\begin{cases} \tau_{11} = 2\lambda\nu \left(\frac{\partial u_1}{\partial y}\right)^2 = 32\lambda\nu n^2(2y - 1)^{4n-2} \\ \tau_{12} = \nu \frac{\partial u_1}{\partial y} = -4\nu n(2y - 1)^{2n-1} \\ \tau_{22} = 0 \end{cases} \quad (5.36)$$

où ν est le coefficient de viscosité et λ le temps de relaxation.

Dans les prochaines sections, nous décrivons les résultats pour $n = 5$ en posant successivement la variable λ égale à 0.0 et à 1.0.

5.5.5 Résolution du problème de la couche limite : $\lambda = 0.0$

Pour ce problème, nous avons effectué 3 séquences d'adaptation. Les maillages obtenus sont nommés respectivement ADAPT1, ADAPT2 et ADAPT3. Les caractéristiques des maillages sont présentées dans le tableau 5.7 tandis que les résultats sont indiqués aux

tableaux 5.8 et 5.9. Pour fin de comparaison, nous avons généré un maillage régulier possédant environ le même nombre de sommets que ADAPT3 sur lequel nous avons résolu le problème discret. Nous y remarquons d'abord que pour les variables u et p , l'erreur diminue ou reste constante. Nous jugeons ces résultats satisfaisant puisque l'estimateur ne tient pas compte de ces variables.

Pour τ , nous remarquons que l'erreur en norme infinie sur le maillage régulier REG est deux fois plus élevé que sur le maillage adapté ADAPT3. Par contre, elle est du même ordre de grandeur pour la norme l_2 . Globalement, l'erreur sur τ décroît à chaque nouvelle adaptation ce qui indique que l'estimateur capte bien l'erreur commise sur τ .

Les maillages obtenus sont très satisfaisants car la qualité moyenne des éléments est toujours au dessus de 0.8. La longueur désirée fut fixée à $l_d = 4.0$. De plus, l'adaptation semble converger rapidement vers le maillage optimal. En effet, l'augmentation la plus importante du nombre de sommets se produit à la première adaptation. Le nombre de sommets augmente alors de l'ordre de 5 fois. À la troisième adaptation, cette augmentation est très petite bien que l'erreur sur τ diminue de façon considérable. Nous pouvons voir les maillages adaptés ainsi que le maillage régulier REG à la figure 5.7.

	Maillages				
Qualité	ORIGINAL	ADAPT1	ADAPT2	ADAPT3	REG
Q_K moyenne	0.968861	0.872692	0.836043	0.819019	0.774246
Q_K minimum	0.708772	0.147909	0.0844497	0.0446155	0.361399
Q_K maximum	0.999993	0.999996	0.999907	0.999995	0.999994
Nombre d'éléments	938	4989	7097	7353	8266
Nombre de sommets	510	2709	3999	4272	4286

TAB. 5.7 – Caractéristiques des maillages générés pour le problème de couche limite : Cas $\lambda = 0.0$.

	Maillages				
Erreur	ORIGINAL	ADAPT1	ADAPT2	ADAPT3	REG
$\ (\mathbf{u}_h)_x - (\pi_h \mathbf{u})_x\ _2$	0.0044029	0.000144345	0.00140179	0.00127072	0.000259434
$\ (\mathbf{u}_h)_x - (\pi_h \mathbf{u})_x\ _\infty$	0.00928707	0.00132453	0.00744277	0.00605966	0.00297134
$\ (\mathbf{u}_h)_y - (\pi_h \mathbf{u})_y\ _2$	0.00173675	7.652e-05	0.000420005	0.000400395	0.000127219
$\ (\mathbf{u}_h)_y - (\pi_h \mathbf{u})_y\ _\infty$	0.00504513	0.00061053	0.00185351	0.0011729	0.00122421
$\ p_h - \pi_h p\ _2$	0.107147	0.0260772	0.0440234	0.029469	0.0344043
$\ p_h - \pi_h p\ _\infty$	0.926667	0.689562	0.498937	0.342752	1.0351
$\ (\boldsymbol{\tau}_h)_{11} - (\pi_h \boldsymbol{\tau})_{11}\ _2$	0.0752552	0.01004	0.0209635	0.0198917	0.0238859
$\ (\boldsymbol{\tau}_h)_{11} - (\pi_h \boldsymbol{\tau})_{11}\ _\infty$	0.290807	0.0995958	0.0943059	0.0781726	0.128576
$\ (\boldsymbol{\tau}_h)_{12} - (\pi_h \boldsymbol{\tau})_{12}\ _2$	2.35458	0.178335	0.294951	0.308181	0.246724
$\ (\boldsymbol{\tau}_h)_{12} - (\pi_h \boldsymbol{\tau})_{12}\ _\infty$	4.92221	0.754238	0.820211	0.612741	1.02566
$\ (\boldsymbol{\tau}_h)_{22} - (\pi_h \boldsymbol{\tau})_{22}\ _2$	0.0752551	0.0100399	0.0209635	0.0198917	0.0238859
$\ (\boldsymbol{\tau}_h)_{22} - (\pi_h \boldsymbol{\tau})_{22}\ _\infty$	0.290807	0.0995977	0.0943047	0.0781728	0.128577
Nombre d'éléments	938	4989	7097	7353	8266
Nombre de sommets	510	2709	3999	4272	4286

TAB. 5.8 – Résultats du problème de la couche limite : Cas $\lambda = 0.0$.

	Maillages		
	ADAPT1	ADAPT2	ADAPT3
Longueur calculée l_c	4.09336	4.0	4.0
Nb raf. d'arêtes	5243	7528	6153
Nb déraf. d'arêtes	3172	6581	6136
Nb ret. d'arêtes	6443	11407	12128
Nb dép. sommets	36702	52090	50671
Dép. sommet Moyen	0.00069	0.000637	0.000642
Dép. sommet Max.	0.0152	0.0137547	0.0144488
Temps d'adaptation (sec.)	339.41	555.7	553.63
Mémoire utilisée (Octets)	8 416 248	12 077 016	12 605 352
Dimension Syst. Lin. (Nb. lignes)	23521	34187	36064

TAB. 5.9 - Données relatives à l'adaptation pour le problème de couche limite : Cas $\lambda = 0.0$.

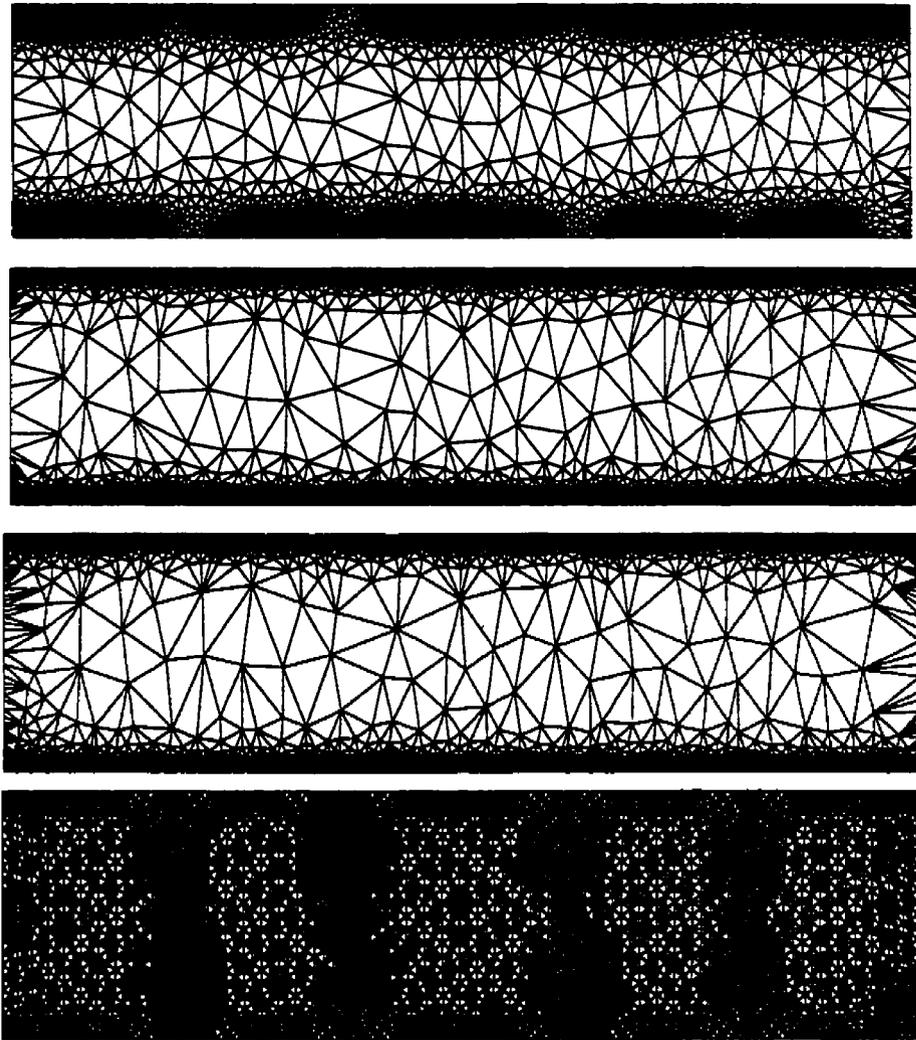


FIG. 5.7 – Maillages ADAPT1, ADAPT2, ADAPT3 et REG (de haut en bas) générés pour le problème de couche limite : Cas $\lambda = 0.0$.

5.5.6 Résolution du problème de la couche limite : $\lambda = 1.0$

De la même manière que le problème précédent, nous allons comparer les résultats d'adaptation sur un maillage régulier, que nous nommerons REG, avec les quatre maillages adaptés qui seront nommés respectivement ADAPT1, ADAPT2, ADAPT3 et ADAPT4. Les caractéristiques de ces maillages sont placées dans le tableau 5.10 tandis que les résultats apparaissent à la figure 5.11. Nous avons fixé la longueur désirée à $l_d = 60.0$. Les autres paramètres concernant les résultats d'adaptation sont disponibles dans le tableau 5.12.

Les maillages sont très satisfaisants car la qualité moyenne est bien en deça de 0.85. L'adaptation a rapidement convergé en deux séquences vers un maillage de 9000 sommets. L'augmentation du nombre de sommets s'est faite d'un facteur 5 pour ADAPT1 et d'un facteur 3.5 pour ADAPT2. Pour les deux autres maillages adaptés, le nombre de sommets a peu varié. De plus, la convergence vers un maillage de 9000 sommets aurait pu être plus rapide si le nombre de divisions n_{div} , défini à la section 2.2.6, avait été plus grand. Nous avons choisi de le fixer à 4.0, ce qui a empêché la valeur l_c d'égaliser la longueur désirée lors de la première itération. En effet, nous remarquons dans le tableau 5.12 que $l_c = 63.9609$ comparativement à $l_d = 60.0$.

Concernant les variables (\mathbf{u}, p) , l'erreur commise est plus petite pour les maillages réguliers qu'adaptés. Les résultats sont quand même jugés très satisfaisants du fait puisque l'estimateur *a posteriori* ne dépend pas de (\mathbf{u}, p) .

Pour $\boldsymbol{\tau}$, nous voyons que l'erreur commise sur les composantes τ_{11} , τ_{12} et τ_{22} diminue de façon significative pour les trois composantes. Elle diminue respectivement d'un facteur 30, 5, 9 en norme l_2 pour τ_{11} , τ_{12} , τ_{22} et respectivement d'un facteur 8, 5, 7 en norme infinie. Cette diminution de l'erreur s'accompagne d'une augmentation de 16 fois du nombre de sommets et d'éléments. La variable la plus difficile à approximer est τ_{11} qui est une fonction polynômiale d'ordre 18. Nous voyons que l'erreur a été grandement diminuée et

elle est 4 fois plus petite pour un maillage régulier (REG). Ce gain appréciable en calculs justifie l'utilisation de l'adaptation de maillages. Pour τ_{12} et τ_{22} , le gain par rapport au maillage régulier est moindre. Pour ce problème, l'estimateur dépend fortement de τ_{11} et c'est pourquoi le gain en calcul est inférieur sur les autres variables. En conclusion, nous en ressortons gagnant d'adopter une approche qui utilise l'adaptation de maillages. La solution obtenue est bien meilleure et ce, pour un coût équivalent en calculs.

	Maillages					
Qualité	ORIGINAL	ADAPT1	ADAPT2	ADAPT3	ADAPT4	REG
Q_K moyenne	0.968861	0.876996	0.879494	0.868466	0.874839	0.86931
Q_K minimum	0.708772	0.0662877	0.0423391	0.0258299	0.0222528	0.361399
Q_K maximum	0.999993	0.999984	0.999996	0.999999	0.999999	0.999996
Nombre d'éléments	938	4859	16650	14714	15266	16620
Nombre de sommets	510	2631	8952	8119	8438	8477

TAB. 5.10 – Caractéristiques des maillages générés pour le problème de couche limite : Cas $\lambda = 1.0$.

Erreur	Maillages					
	ORIGINAL	ADAPT1	ADAPT2	ADAPT3	ADAPT4	REG
$\ (\mathbf{u}_h)_x - (\pi_h \mathbf{u})_x\ _2$	0.0044029	0.000173964	0.000692041	0.00305793	0.00505797	0.000176695
$\ (\mathbf{u}_h)_x - (\pi_h \mathbf{u})_x\ _\infty$	0.00928707	0.00222157	0.00200933	0.0132207	0.01955	0.00128309
$\ (\mathbf{u}_h)_y - (\pi_h \mathbf{u})_y\ _2$	0.00173675	6.79346e-05	0.000274622	0.00113193	0.00152977	7.59309e-05
$\ (\mathbf{u}_h)_y - (\pi_h \mathbf{u})_y\ _\infty$	0.00504513	0.000481142	0.000799058	0.00326754	0.00765998	0.00056176
$\ p_h - \pi_h p\ _2$	0.106558	0.0448697	0.0289867	0.053706	0.0801259	0.459713
$\ p_h - \pi_h p\ _\infty$	0.93	0.313157	0.401289	0.468799	0.538727	0.465532
$\ (\boldsymbol{\tau}_h)_{11} - (\pi_h \boldsymbol{\tau})_{11}\ _2$	262.345	40.3808	6.44523	7.68582	8.96877	40.2396
$\ (\boldsymbol{\tau}_h)_{11} - (\pi_h \boldsymbol{\tau})_{11}\ _\infty$	537.689	240.915	77.2083	84.6839	64.6579	265.226
$\ (\boldsymbol{\tau}_h)_{12} - (\pi_h \boldsymbol{\tau})_{12}\ _2$	3.3249	0.386251	0.288769	0.557769	0.688425	0.397736
$\ (\boldsymbol{\tau}_h)_{12} - (\pi_h \boldsymbol{\tau})_{12}\ _\infty$	8.00753	2.9706	0.769081	1.48231	2.8195	2.70598
$\ (\boldsymbol{\tau}_h)_{22} - (\pi_h \boldsymbol{\tau})_{22}\ _2$	0.0637152	0.00744834	0.00124849	0.00473413	0.00706841	0.00792376
$\ (\boldsymbol{\tau}_h)_{22} - (\pi_h \boldsymbol{\tau})_{22}\ _\infty$	0.391694	0.108695	0.0368356	0.0557805	0.0570164	0.0908239
Nombre d'éléments	938	4859	16650	14714	15266	16620
Nombre de sommets	510	2631	8952	8119	8438	8477

TAB. 5.11 – Résultats du problème de la couche limite : Cas $\lambda = 1.0$.

	Maillages			
	ADAPT1	ADAPT2	ADAPT3	ADAPT4
Longueur calculée l_c	63.9609	60.0	60.0	60.0
Nb raf. d'arêtes	4669	14288	4971	5393
Nb déraf. d'arêtes	2633	8172	5907	5103
Nb ret. d'arêtes	5381	15979	9261	9810
Nb dép. sommets	34279	96289	89454	91038
Dép. sommet Moyen	0.000718	0.000449	0.0005471	0.000448
Dép. sommet Max.	0.018816	0.153824	0.0329252	0.015427
Temps d'adaptation (sec.)	317.84	1057.53	822.46	821.5
Mémoire utilisée (Octets)	8 191 608	28 023 864	24 915 048	25 860 120
Dimension Syst. Lin. (Nb. lignes)	22871	78058	70021	72720

TAB. 5.12 – Données relatives à l'adaptation pour le problème de couche limite : Cas $\lambda = 1.0$.

Nous pouvons voir aux figures 5.8 et 5.9 les maillages générés. De plus, nous avons illustré certaines solutions obtenues sur le maillage ADAPT4 dont τ_{11} , τ_{12} , u_x et p .

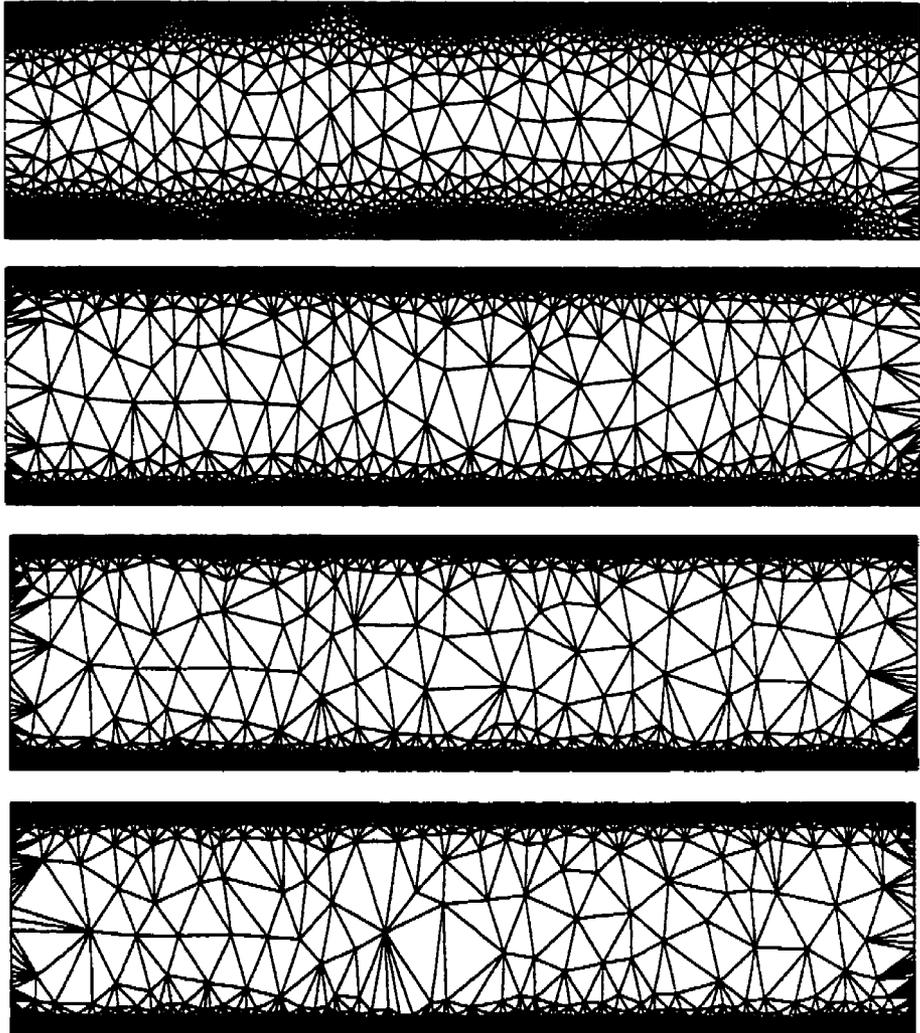


FIG. 5.8 – Maillages ADAPT1, ADAPT2, ADAPT3 et ADAPT4 (de haut en bas) générés pour le problème de couche limite : Cas $\lambda = 1.0$.

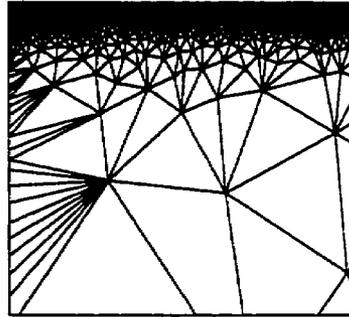


FIG. 5.9 – Agrandissement du maillage ADAPT4 (coin supérieur gauche) généré pour le problème de couche limite : Cas $\lambda = 1.0$.

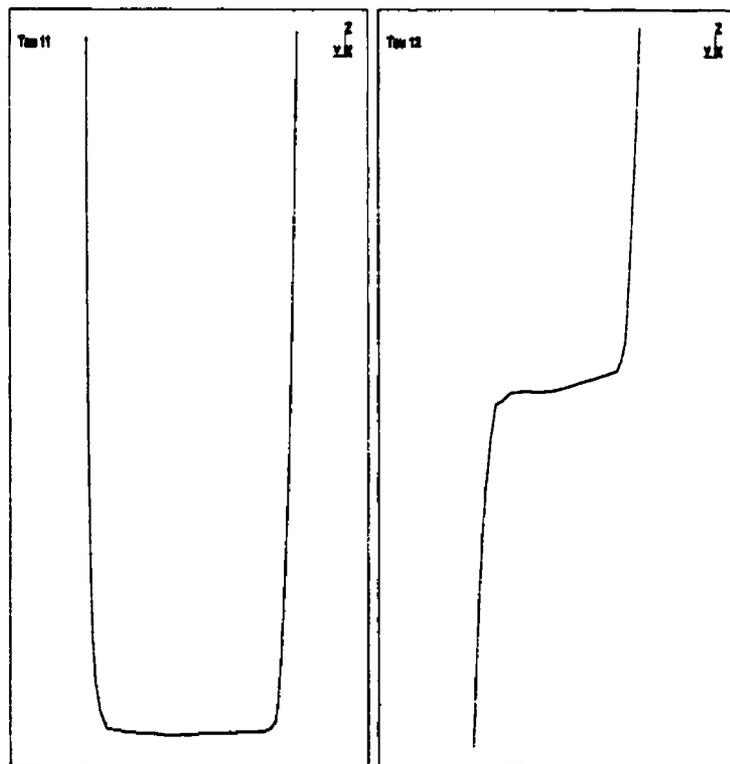


FIG. 5.10 – Coupe de la solution τ_{11} et τ_{12} en $x=2.5$: Cas $\lambda = 1.0$ du problème de couche limite.

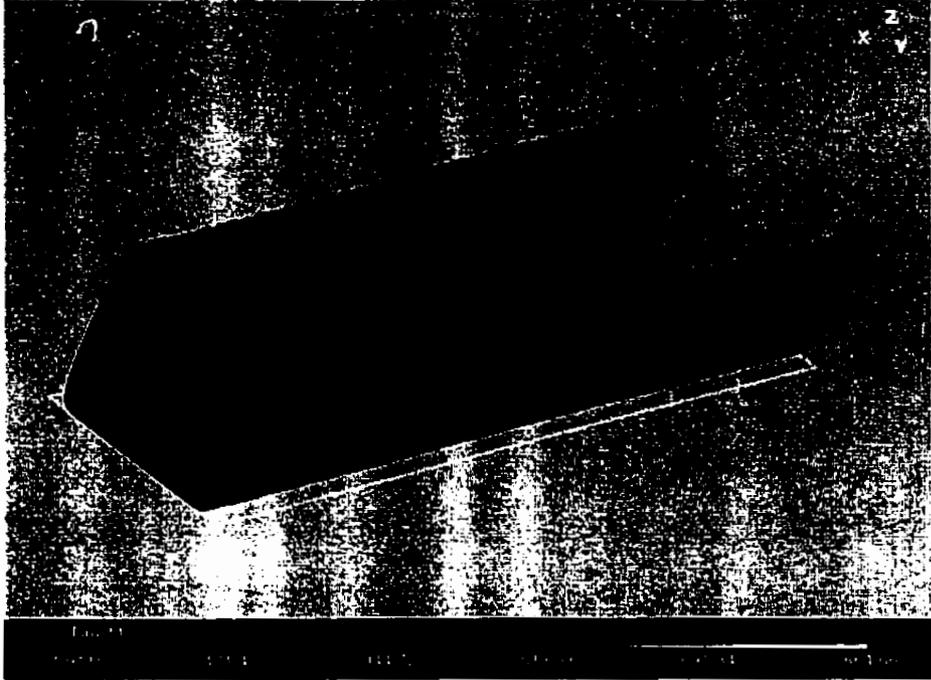


FIG. 5.11 – Solution τ_{11} : Cas $\lambda = 1.0$ du problème de couche limite.

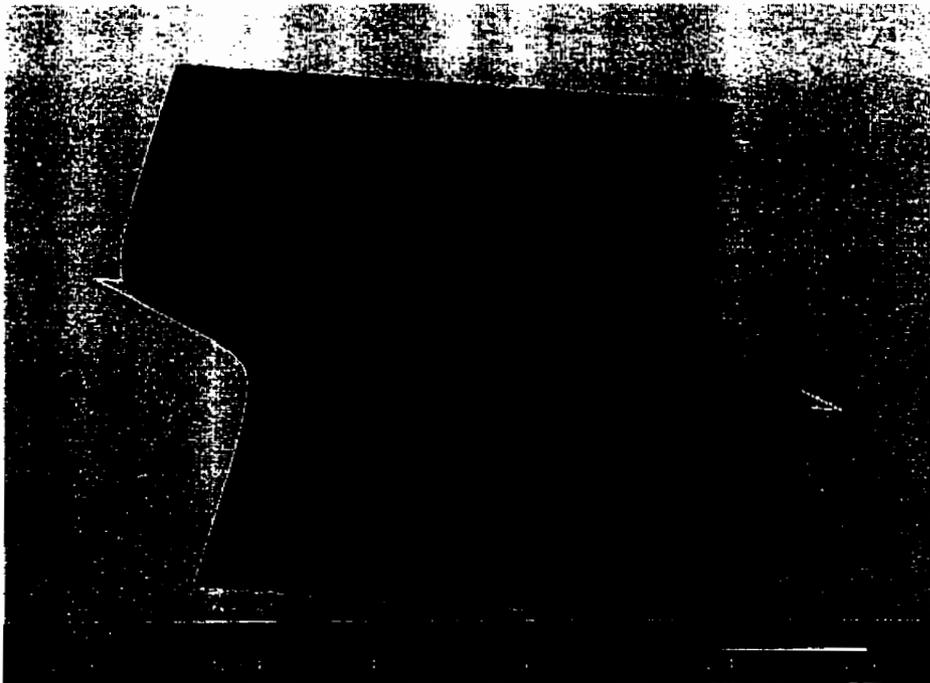


FIG. 5.12 – Solution τ_{12} : Cas $\lambda = 1.0$ du problème de couche limite.

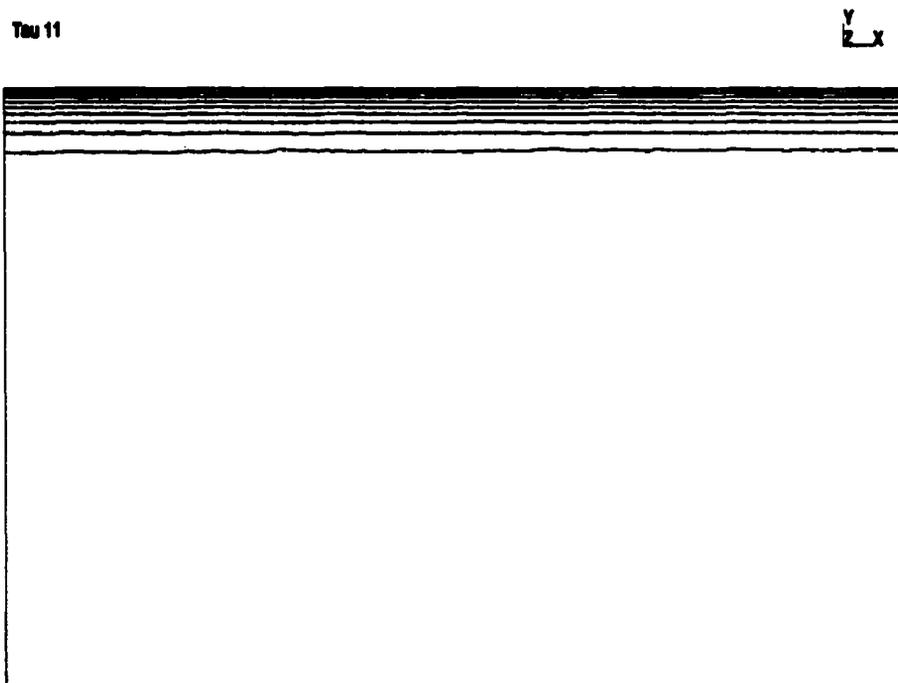


FIG. 5.13 – Courbes de niveau de τ_{11} , coin supérieur gauche : Cas $\lambda = 1.0$ du problème de couche limite.

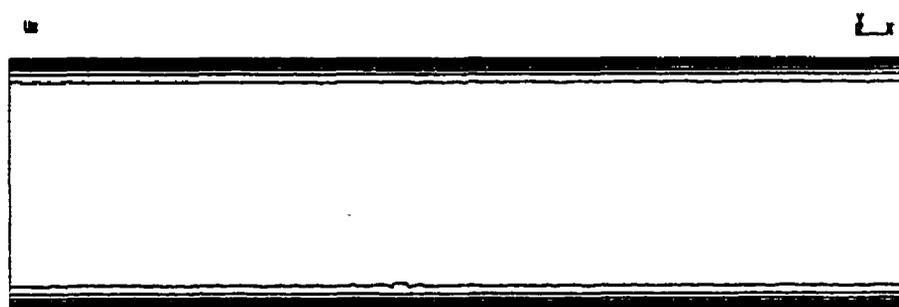


FIG. 5.14 – Courbes de niveau de u_x : Cas $\lambda = 1.0$ du problème de couche limite.

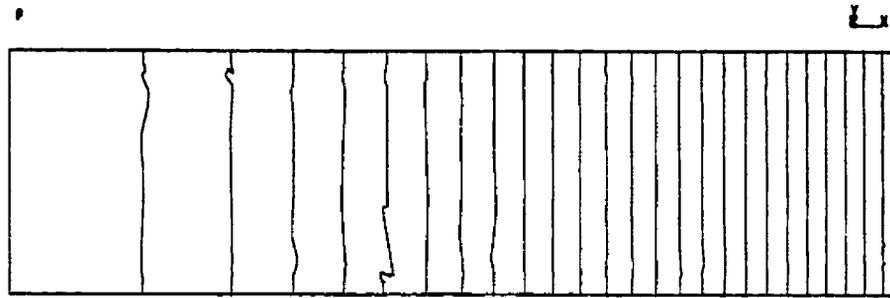


FIG. 5.15 – Courbes de niveau de p : Cas $\lambda = 1.0$ du problème de couche limite.

5.5.7 Problème de l'écoulement autour d'un cylindre

Nous allons maintenant aborder l'étude du problème de l'écoulement autour d'un cylindre. Contrairement au problème de Poiseuille et de la couche limite, nous ne connaissons pas la solution analytique qui décrit le comportement du fluide. L'écoulement étudié est en fait un écoulement de Poiseuille auquel nous avons ajouté un obstacle circulaire de façon à obstruer le canal. L'obstacle en question possède un rayon $R = 0.25$ et est situé à 1 de l'entrée et à égale distance des parois du canal. La géométrie est représentée à la figure 5.16. Ici, nous faisons les mêmes hypothèses que celles faites pour le problème de Poiseuille (Sect. 5.5.1).

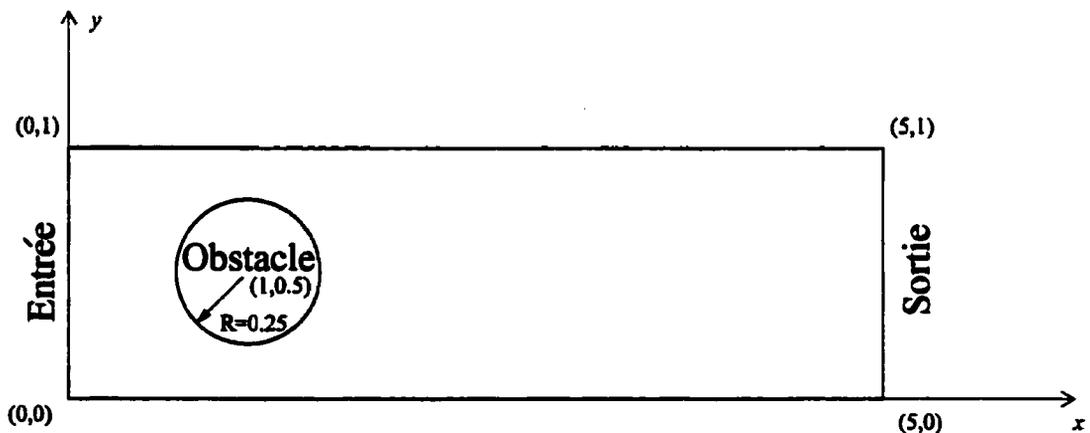


FIG. 5.16 – Géométrie du canal avec obstacle.

Pour les conditions aux limites, nous imposons pour u à l'entrée, la solution exacte

du problème de Poiseuille donnée par l'équation (5.33). À la sortie, nous posons que la composante verticale de la vitesse soit nulle i.e. $\mathbf{u} = (u_x, 0)$. Par ailleurs, comme nous avons fait l'hypothèse que le fluide adhère à la surface du canal, nous devons imposer que le fluide ait une vitesse nulle $\mathbf{u} = (0, 0)$ sur les parois du canal et de l'obstacle. Pour la résolution du tenseur $\boldsymbol{\tau}$ des contraintes supplémentaires (problème de Maxwell), nous imposons à l'entrée la solution exacte du problème de Poiseuille (5.34) où $\nu = 1.0$.

Nous décrivons dans les sections suivantes, la résolution du problème de l'écoulement autour d'un cylindre dans le cas newtonien (Sect. 5.5.8) et dans le cas viscoélastique (Sect. 5.5.9).

5.5.8 Résolution de l'écoulement autour d'un cylindre : $\lambda = 0.0$

Tout d'abord, nous avons effectué trois adaptations dans le cas newtonien. Le maillage de base est illustré à la figure 5.17 tandis que les maillages obtenus ADAPT1, ADAPT2 et ADAPT3 sont illustrés à la figure 5.18 et un agrandissement du maillage final ADAPT3 à la figure 5.19. Les caractéristiques de ces maillages sont présentés au tableau 5.13 et les résultats d'adaptation au tableau 5.14.

Nous remarquons d'abord que l'adaptation de maillage fait un excellent travail en ajoutant tous les sommets à la première séquence. De plus, la qualité moyenne des éléments est toujours supérieure à 0.88 ce qui dénote des maillages bien construits. La longueur désirée qui a été choisie à $l_d = 3.0$ n'a pas été atteinte à la première passe d'adaptation. Il était donc normal de s'attendre à une forte augmentation du nombre de sommets pour faire diminuer l'erreur. Nous voyons que le nombre d'opérations d'adaptation a été de loin plus important lors de la première adaptation.

Nous observons que le maillage est raffiné dans les zones où le tenseur $\boldsymbol{\tau}$ varie le plus. De plus, les solutions sont régulières ce qui indique que le maillage adapté tient compte

efficacement de l'erreur commise sur τ . Pour la variable u , le champs de vitesse contourne l'obstacle de façon régulière et symétrique. Les figures qui suivent illustrent les résultats obtenus sur le maillage adapté ADAPT3.

Qualité	Maillages			
	ORIGINAL	ADAPT1	ADAPT2	ADAPT3
Q_K moyenne	0.956554	0.883398	0.880366	0.886484
Q_K minimum	0.552973	0.220566	0.16455	0.149985
Q_K maximum	0.999984	0.999956	0.999988	0.999988
Nombre d'éléments	1044	5740	6277	6407
Nombre de sommets	572	3028	3354	3432

TAB. 5.13 – Caractéristiques des maillages générés pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 0.0$.

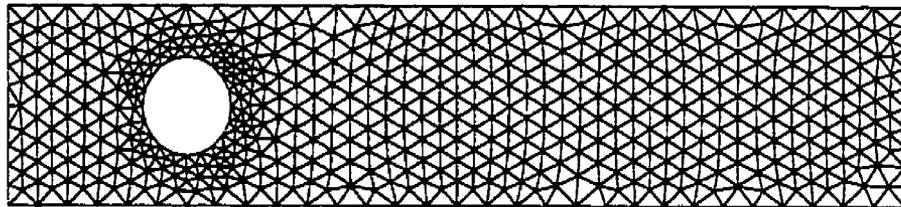


FIG. 5.17 – Maillage ORIGINAL utilisé pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 0.0$.

	Maillages		
	ADAPT1	ADAPT2	ADAPT3
Longueur calculée l_c	3.16411	3.0	3.0
Nb raf. d'arêtes	4608	1831	1282
Nb déraf. d'arêtes	2222	1546	1222
Nb ret. d'arêtes	5080	3273	2838
Nb dép. sommets	35181	36856	36696
Dép. sommet Moyen	0.000688	0.000544	0.000492633
Dép. sommet Max.	0.029	0.016316	0.0178659
Temps d'adaptation (sec.)	342.54	332.76	321.91
Mémoire utilisée (Octets)	9 619 800	10 550 472	10 775 112
Dimension Syst. Lin. (Nb. lignes)	26620	29324	29974

TAB. 5.14 – Données relatives à l'adaptation pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 0.0$.

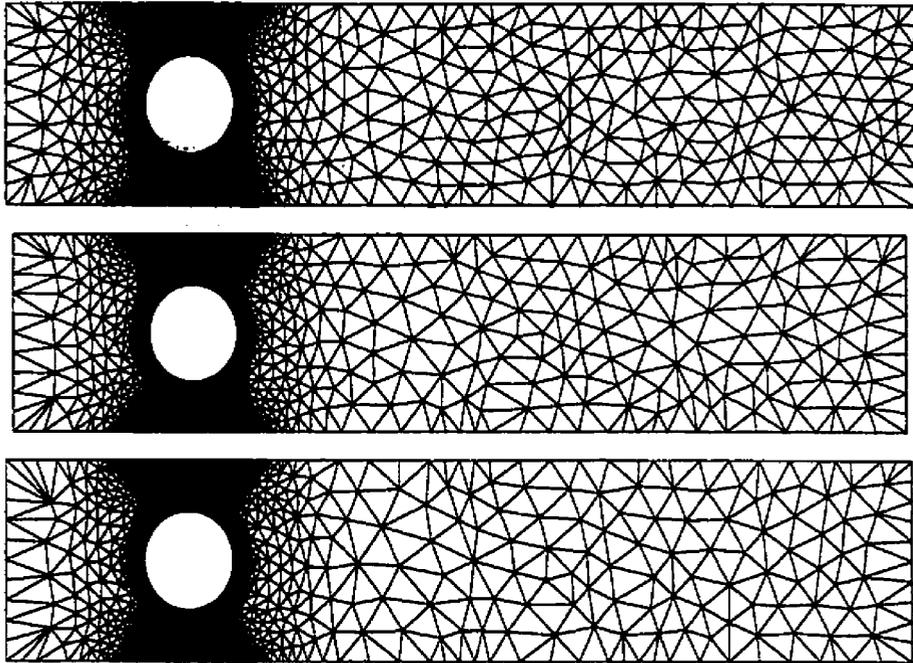


FIG. 5.18 – Maillages ADAPT1, ADAPT2 et ADAPT3 (de haut en bas) générés pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 0.0$.

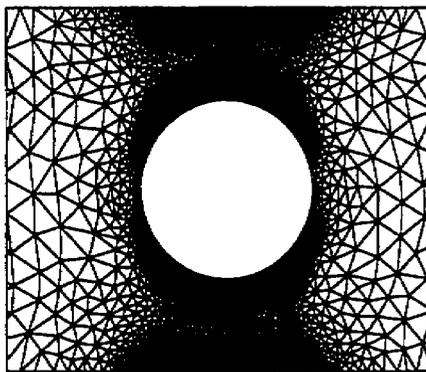


FIG. 5.19 – Agrandissement du maillage ADAPT3 (aux alentours de l'obstacle) généré pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 0.0$.

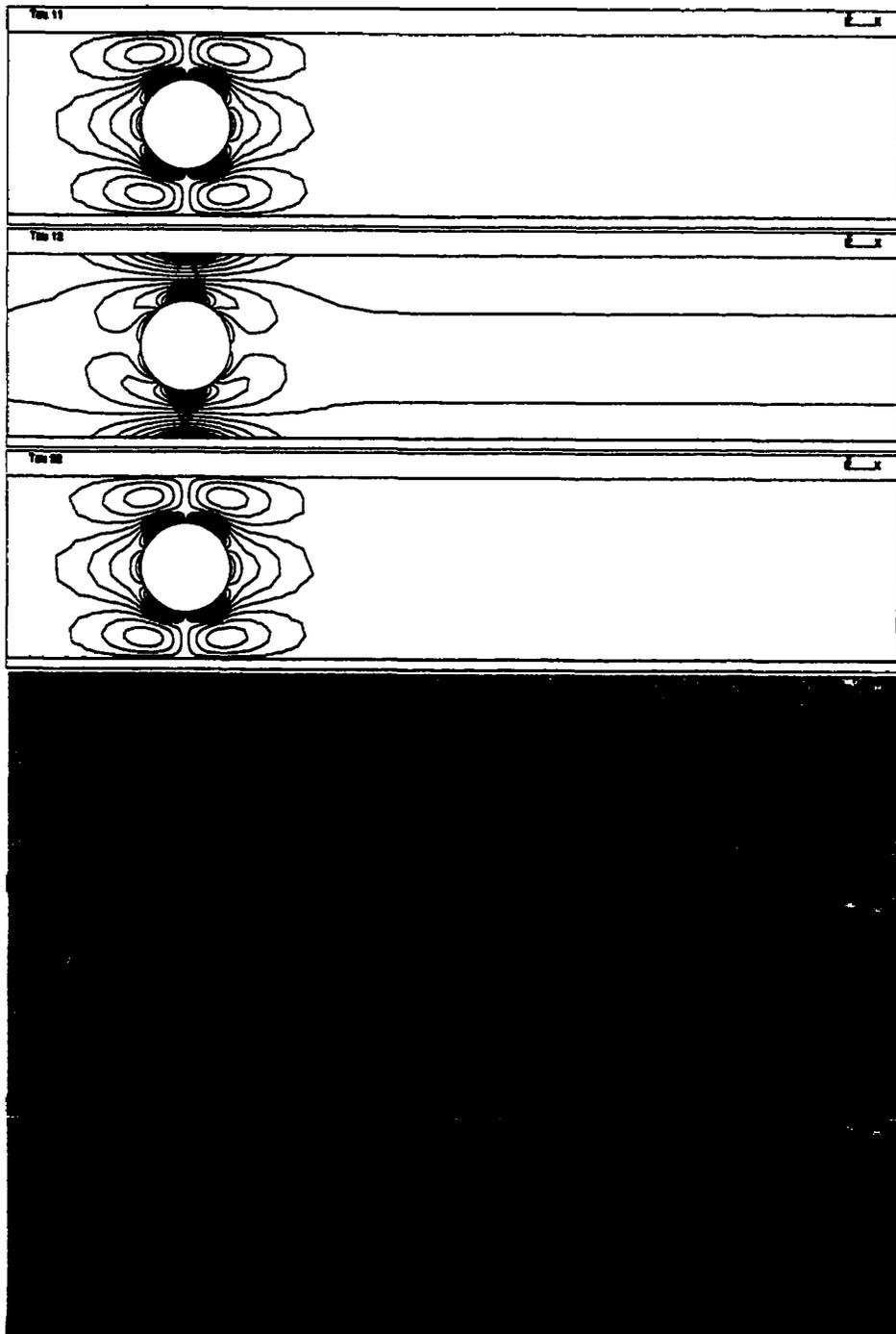


FIG. 5.20 – De haut en bas, les courbes de niveau des variables τ_{11} , τ_{12} et τ_{22} et par la suite les graphes des variables τ_{11} , τ_{12} et τ_{22} sur ADAPT3 pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 0.0$

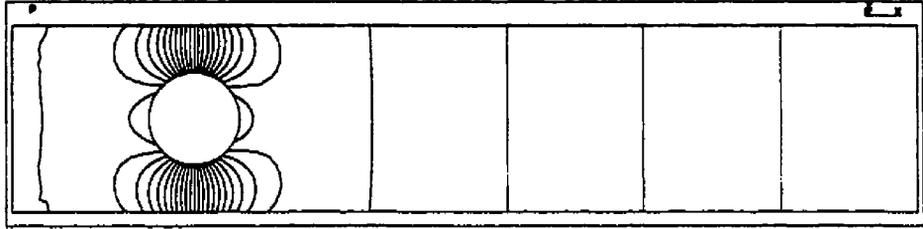


FIG. 5.21 – Courbes de niveau de p sur ADAPT3 pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 0.0$.



FIG. 5.22 – Graphe de la variable u sur ADAPT3 autour de l'obstacle pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 0.0$.

5.5.9 Résolution de l'écoulement autour d'un cylindre : $\lambda = 1.0$

Dans le cas viscoélastique, nous avons effectué quatre adaptations et les maillages obtenus ADAPT1, ADAPT2, ADAPT3 et ADAPT4 sont illustrés aux figures 5.23 et 5.24. Les caractéristiques de ces maillages sont présentés au tableau 5.15 et les résultats d'adaptation au tableau 5.16.

Nous remarquons d'abord que l'adaptation de maillage s'est très bien comportée conservant la qualité moyenne des éléments qui est toujours supérieure à 0.88. Nous voyons aussi que le nombre de sommets a augmenté durant les deux premières séquences d'adaptations et a diminué par la suite pour rester stable à 4200 sommets. Ce qui démontre que les algorithmes d'adaptation ont raffiné jusqu'à obtenir un maillage (ADAPT2) sur lequel la précision voulue était atteinte. À noter que la longueur désirée $l_d = 200.0$ n'a pas été atteinte à la première adaptation et c'est pourquoi il s'est ajouté un grand nombre de sommets. Or, par la suite, ces algorithmes ont déraffiné le maillage pour conserver cette tolérance tout en minimisant le nombre de sommets. Nous voyons ici tout la force de l'adaptation de maillage qui permet d'optimiser les maillages.

Nous voyons que le nombre d'opérations d'adaptation a été de loin plus important lors des deux premières adaptations. Il est important de noter que le nombre de raffinements, de déraffinements et retournement d'arêtes est très petit pour la dernière adaptation et démontre la convergence rapide vers le maillage optimal.

Les résultats concernant les variables discrètes obtenus sur le maillage final ADAPT4 sont illustrés aux figures 5.25 à 5.28. De ces solutions, nous pouvons voir que la variable τ présente une singularité près de l'obstacle. Nonobstant, les solutions sont régulières et symétriques ce qui montre que le schéma de stabilisation SU est efficace. De plus, nous voyons que l'adaptation de maillages a raffiné dans les régions où τ admet de grandes variations i.e. dans la zone près de l'obstacle et dans la zone de trainée que présente la solution. Si l'on se réfère aux travaux de Michael Renardy sur le comportement asymp-

totique du tenseur des contraintes pour ce type d'écoulement [13], les résultats obtenus viennent confirmer les conclusions théoriques sur la présence d'une trainée et d'une forte singularité au voisinage de l'obstacle. En fait, l'auteur identifie trois zones classées par ordre d'intensité pour les contraintes : le long de la frontière du cylindre, dans la trainée, surtout la partie initiale et près du cylindre. Les figures illustrent bien ce résultat.

	Maillages				
Qualité	ORIGINAL	ADAPT1	ADAPT2	ADAPT3	ADAPT4
Q_K moyenne	0.956554	0.913816	0.908501	0.880693	0.884836
Q_K minimum	0.552973	0.184638	0.0809799	0.0120682	0.00841453
Q_K maximum	0.999984	0.99997	0.999992	0.999979	0.999985
Nombre d'éléments	1044	7629	14213	8121	8040
Nombre de sommets	572	3981	7325	4282	4242

TAB. 5.15 – Caractéristiques des maillages générés pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 1.0$.

	Maillages			
	ADAPT1	ADAPT2	ADAPT3	ADAPT4
Longueur calculée l_c	248.815	200.0	200.0	200.0
Nb raf. d'arêtes	5215	6751	213	284
Nb déraf. d'arêtes	1831	3449	3261	325
Nb ret. d'arêtes	4067	6685	1871	1033
Nb dép. sommets	49234	92915	55774	46457
Dép. sommet Moyen	0.00044	0.00032	0.000533	0.000298363
Dép. sommet Max.	0.01103	0.01186	0.013025	0.00601452
Temps d'adaptation (sec.)	394.08	709.09	393.69	301.13
Mémoire utilisée (Octets)	12 754 104	23 694 648	13 608 456	13 474 680
Dimension Syst. Lin. (Nb. lignes)	35163	65051	37652	37290

TAB. 5.16 – Données relatives à l'adaptation pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 1.0$.

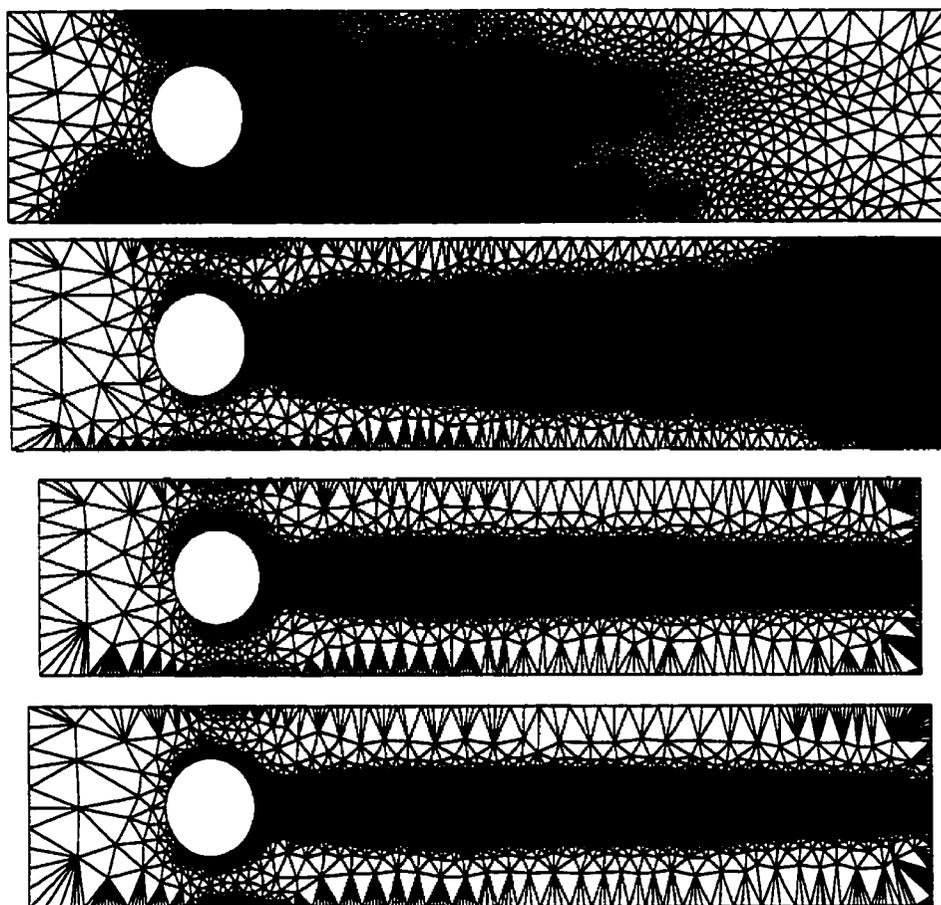


FIG. 5.23 – Maillages ADAPT1, ADAPT2, ADAPT3 et ADAPT4 (de haut en bas) générés pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 1.0$.

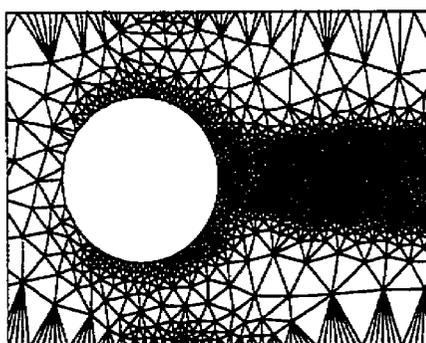


FIG. 5.24 – Agrandissement du maillage ADAPT4 (aux alentours de l'obstacle) généré pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 1.0$.

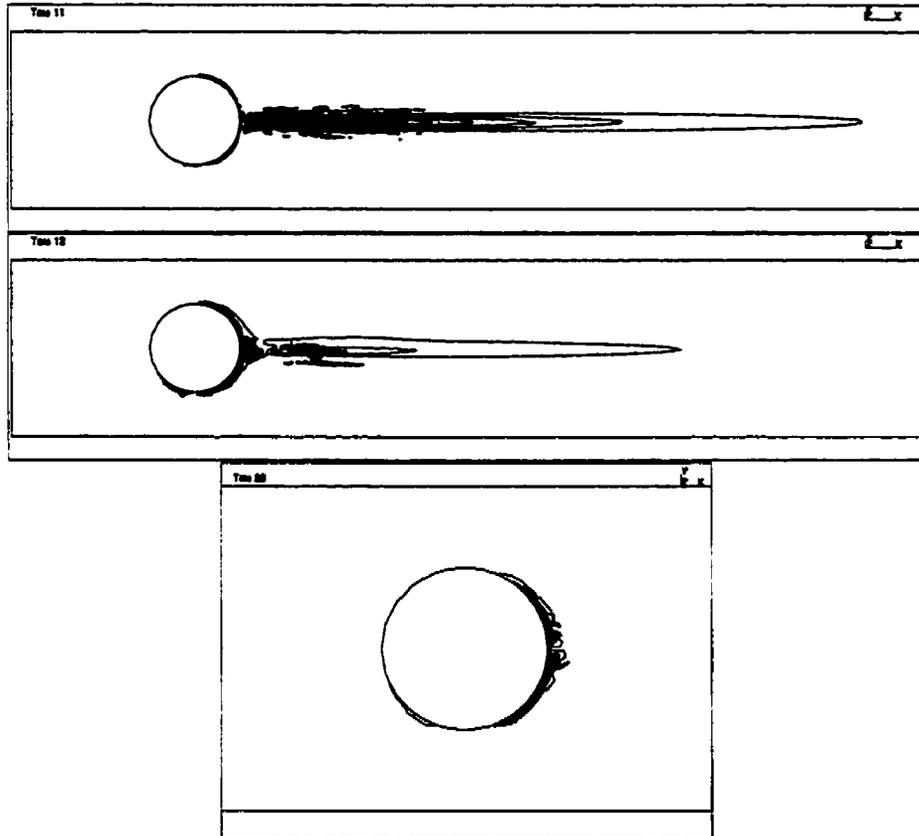


FIG. 5.25 - Courbes de niveau des variables τ_{11} , τ_{12} et τ_{22} (de haut en bas) sur ADAPT4 pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 1.0$.

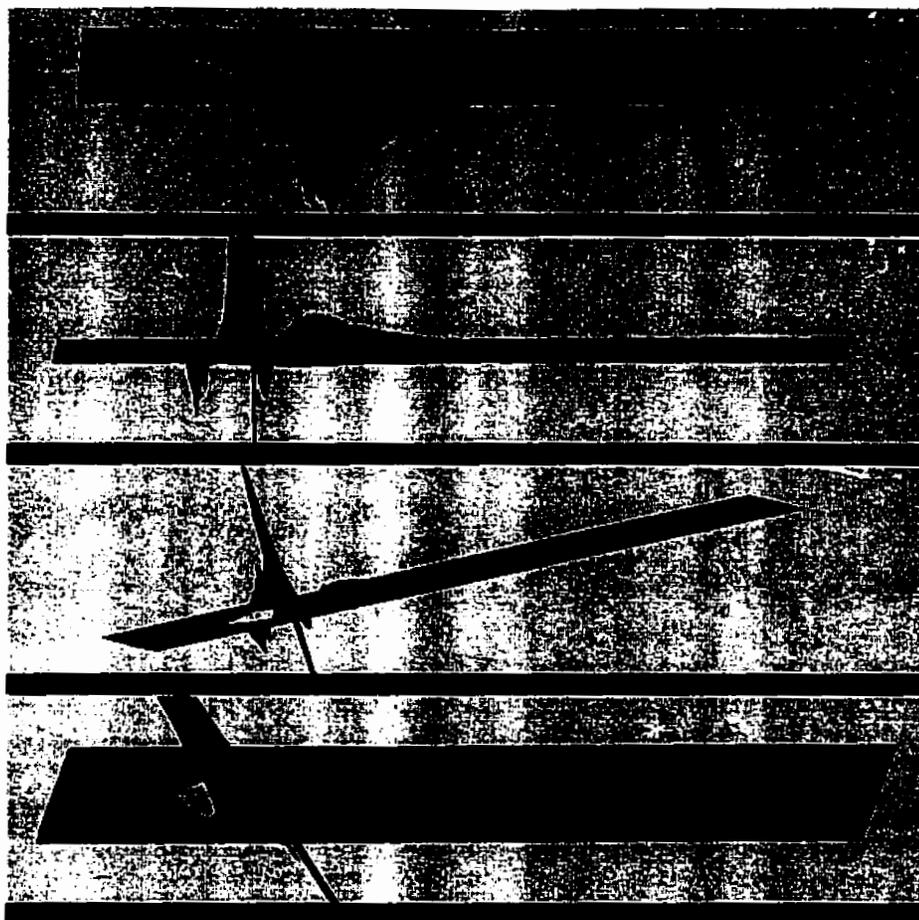


FIG. 5.26 – Graphe des variables τ_{11} , τ_{12} , τ_{12} et τ_{22} (de haut en bas) sur ADAPT4 pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 1.0$.

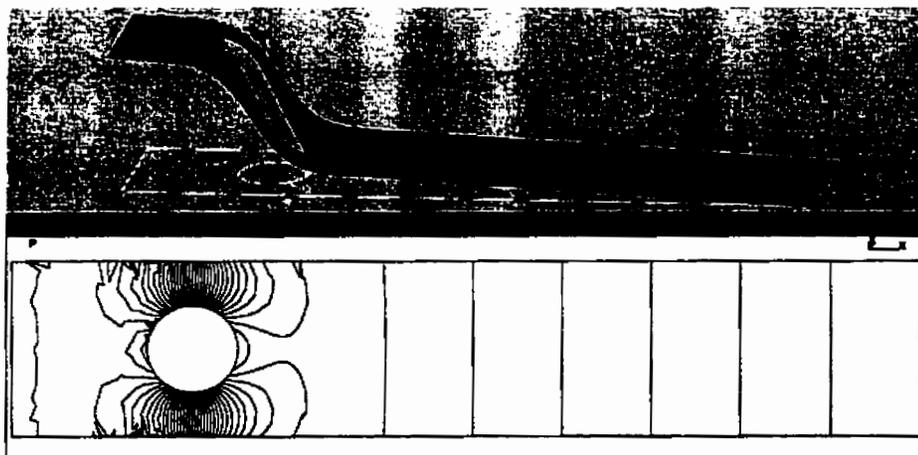


FIG. 5.27 – Graphe et courbes de niveau de la variable p sur ADAPT4 pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 1.0$.

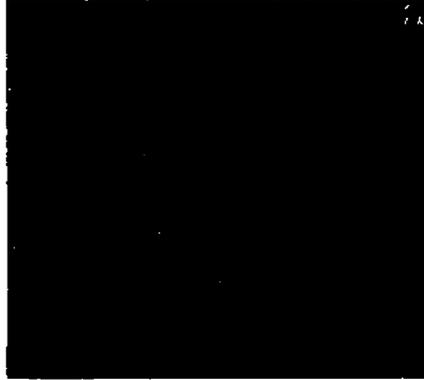


FIG. 5.28 – Graphe de la variable u sur ADAPT4 autour de l'obstacle pour le problème de l'écoulement autour d'un cylindre : Cas $\lambda = 1.0$.

5.5.10 Problème de la contraction

Le problème de la contraction décrit l'écoulement d'un fluide dans un canal qui subitement se rétrécit d'un facteur 4. Le fluide se déplace dans le canal selon une vitesse ayant un profil parabolique. Nous supposons toujours l'adhérence du fluide à la paroi et l'absence de force externe appliquée au système. La géométrie utilisée est un canal rectangulaire de 4×20 se contractant en un canal rectangulaire de 1×40 dont toutes les arêtes sont droites. La géométrie est représentée à la figure 5.29.

Pour les conditions aux limites, nous imposons pour u à l'entrée la solution exacte du problème de Poiseuille pour un canal de 4×20 :

$$\left\{ \begin{array}{l} u = (u_1, u_2) = (1 - (\frac{y}{2} - 1)^2, 0) \end{array} \right. \quad (5.37)$$

Nous imposons que la composante verticale de la vitesse soit nulle à la sortie i.e. $u = (u_x, 0)$. Par ailleurs, comme nous avons supposé que le fluide adhère à la surface du canal, nous devons imposer que le fluide ait une vitesse nulle $u = (0, 0)$ sur les parois de la contraction.

Pour la résolution du tenseur des contraintes supplémentaires τ (problème de Maxwell), nous imposons seulement à l'entrée la solution exacte du problème de Poiseuille pour un

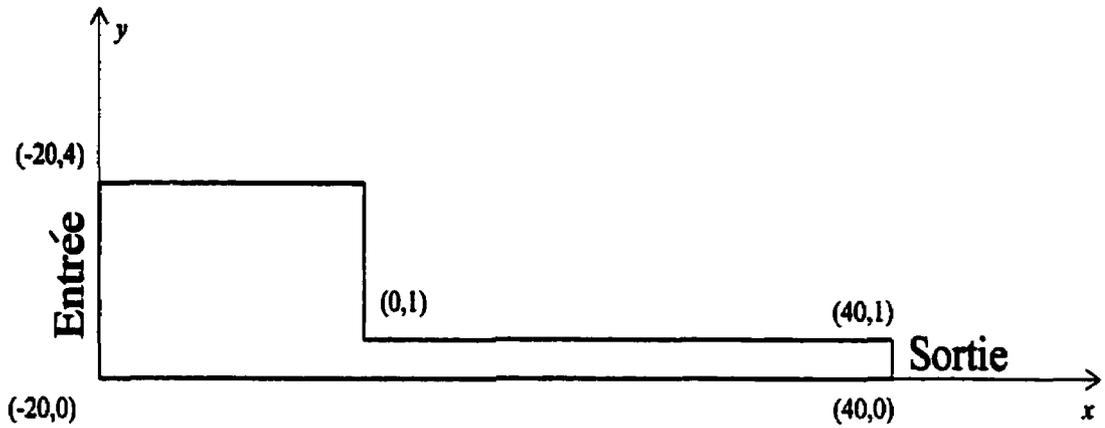


FIG. 5.29 – Géométrie de la contraction.

canal de 4×1 :

$$\begin{cases} \tau_{11} = 2\lambda\nu \left(\frac{\partial u_1}{\partial y}\right)^2 = 2\lambda\nu \left(\frac{y}{2} - 1\right)^2 \\ \tau_{12} = \nu \frac{\partial u_1}{\partial y} = -\nu \left(\frac{y}{2} - 1\right) \\ \tau_{22} = 0 \end{cases} \quad (5.38)$$

où $\nu = 1.0$.

5.5.11 Résolution du problème de la contraction : $\lambda = 0.0$

Dans le cas newtonien, nous avons effectué deux adaptations et les maillages obtenus ADAPT1 et ADAPT2 sont illustrés aux figures 5.31 et 5.32. Les caractéristiques de ces maillages sont présentés au tableau 5.17 et les résultats d'adaptation au tableau 5.18.

Nous remarquons d'abord que l'adaptation de maillage s'est très bien comportée conservant la qualité moyenne des éléments qui est toujours supérieure à 0.87. Nous voyons aussi que le nombre de sommets a augmenté énormément lors de la première adaptation pour se rapprocher du maillage optimal car, dans la deuxième adaptation, le maillage a peu changé et quelques sommets se sont rajoutés. Ceci démontre que les algorithmes d'adaptation tendent à converger rapidement vers un maillage stable. À noter que la longueur désirée fut de $l_d = 3.0$.

On remarque clairement que la contraction possède une singularité au coin (0,1) et que l'adaptation de maillages en a bien tenu compte. On peut voir à la figure 5.32 un agrandissement du maillage qui montre à quel point les algorithmes ont raffiné le maillage. Comme nous avons un critère de longueur minimale l_{min} , cela explique pourquoi toutes les arêtes ont la même longueur aux alentours de la singularité. De plus, on peut voir que les solutions sont très régulières et que plusieurs variables ont une singularité en (0,1) tels que τ_{11} , τ_{12} , τ_{22} et p . Les solutions discrètes obtenus sont illustrées dans les figures qui suivent.



FIG. 5.30 – Maillage ORIGINAL utilisé pour le problème de la contraction : Cas $\lambda = 0.0$.

	Maillages		
Qualité	ORIGINAL	ADAPT1	ADAPT2
Q_K moyenne	0.878633	0.872682	0.886482
Q_K minimum	0.383366	0.166444	0.285675
Q_K maximum	0.999948	0.999998	0.999983
Nombre d'éléments	4936	13297	14964
Nombre de sommets	2755	7111	8089

TAB. 5.17 – Caractéristiques des maillages générés pour le problème de la contraction : Cas $\lambda = 0.0$.

	Maillages	
	ADAPT1	ADAPT2
Longueur calculée l_c	3.0	3.0
Nb raf. d'arêtes	9224	7491
Nb déraf. d'arêtes	5198	6749
Nb ret. d'arêtes	14393	15122
Nb dép. sommets	81069	92198
Dép. sommet Moyen	0.0030466	0.001884
Dép. sommet Max.	0.0403262	0.034160
Temps d'adaptation (sec.)	785.43	900.21
Mémoire utilisée (Octets)	22 352 856	25 217 448
Dimension Syst. Lin. (Nb. lignes)	62147	70371

TAB. 5.18 – Données relatives à l'adaptation pour le problème de la contraction : Cas $\lambda = 0.0$.

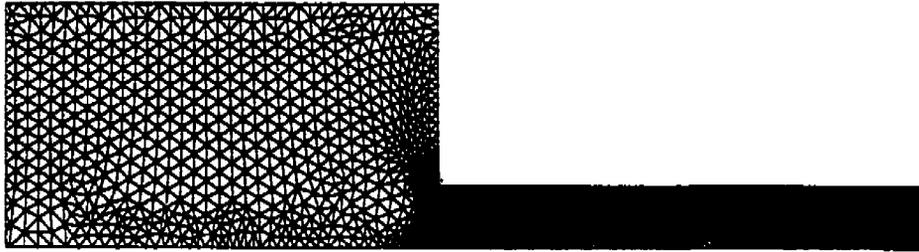


FIG. 5.31 – Maillage ADAPT1 généré pour le problème de la contraction : Cas $\lambda = 0.0$.

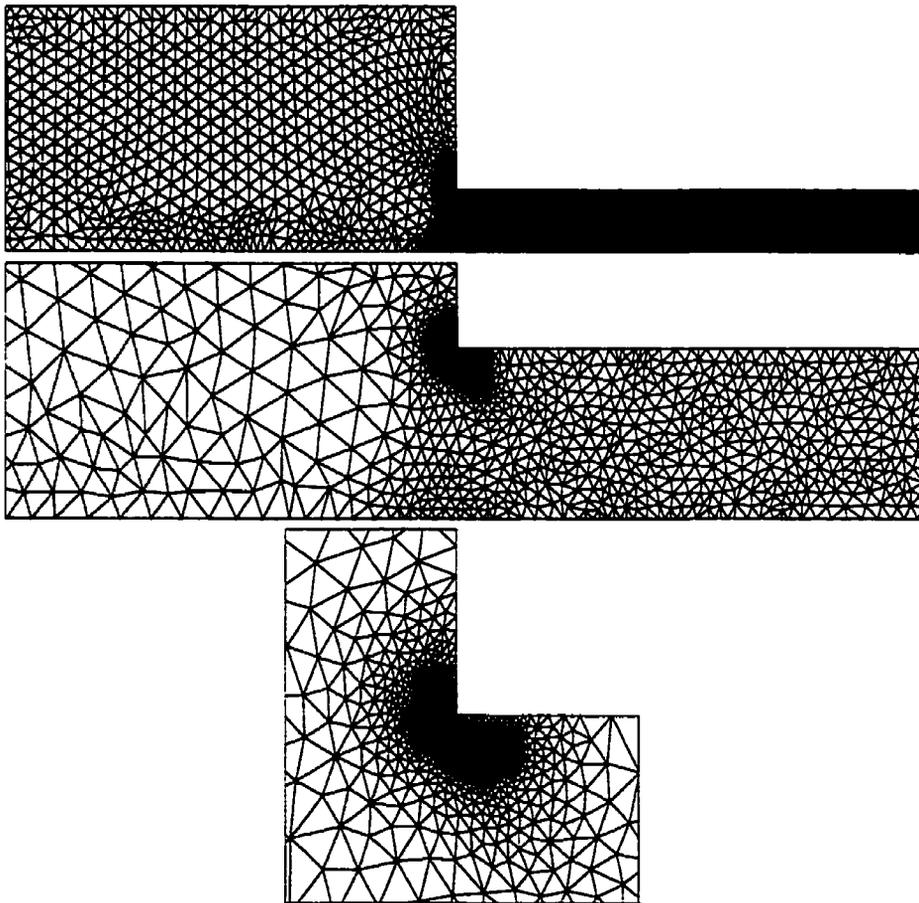


FIG. 5.32 – Maillage ADAPT2, vu sous différents agrandissements, pour le problème de la contraction : Cas $\lambda = 0.0$.

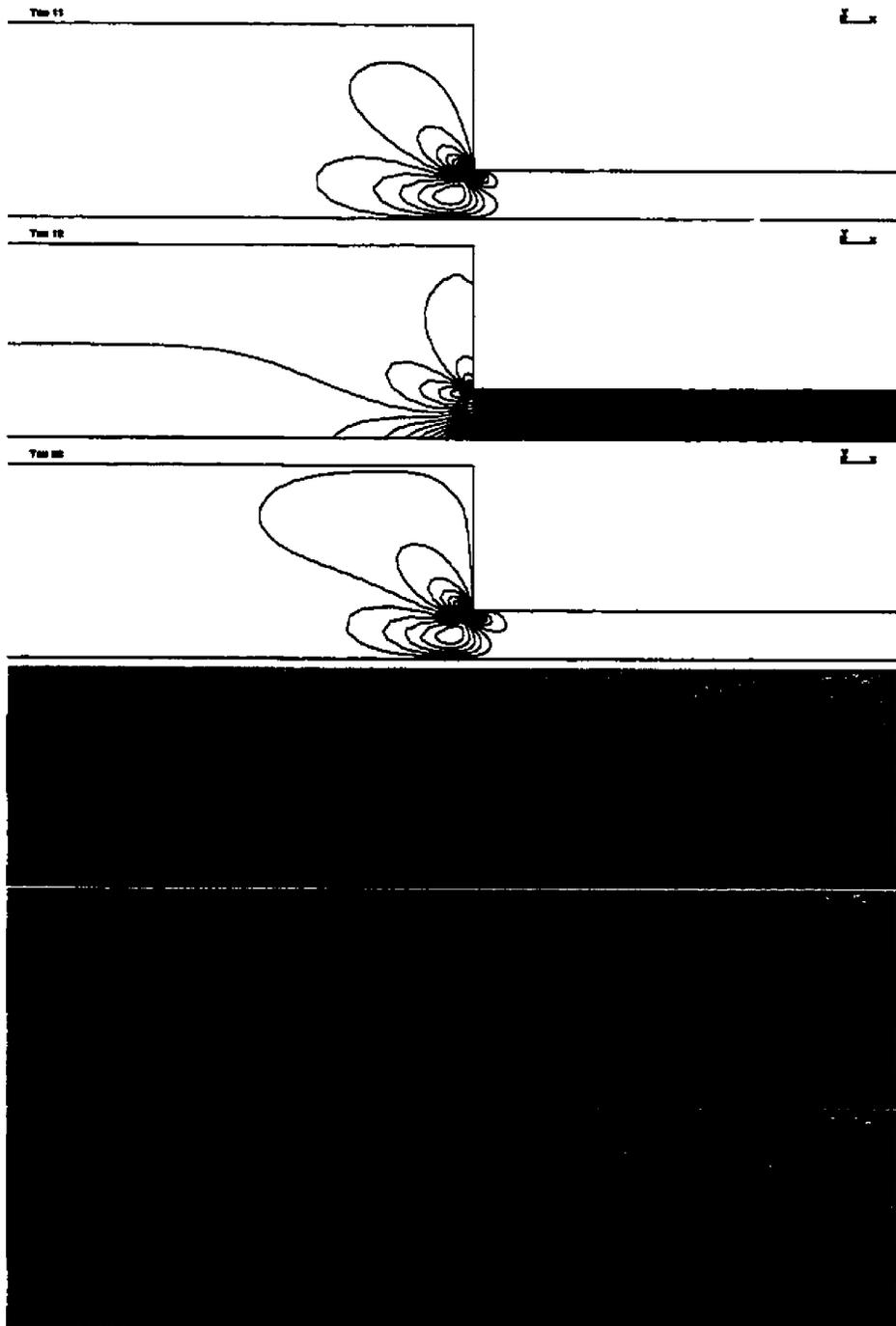


FIG. 5.33 – De haut en bas, les courbes de niveau des variables τ_{11} , τ_{12} et τ_{22} et par la suite les graphes des variables τ_{11} , τ_{12} et τ_{22} sur ADAPT2 pour le problème de la contraction : Cas $\lambda = 0.0$



FIG. 5.34 – Graphe de la variables p selon 2 vues différentes sur ADAPT2 pour le problème de la contraction : Cas $\lambda = 0.0$.



FIG. 5.35 – Graphe de la variable u sur ADAPT2 autour de l'obstacle pour le problème de la contraction : Cas $\lambda = 0.0$.

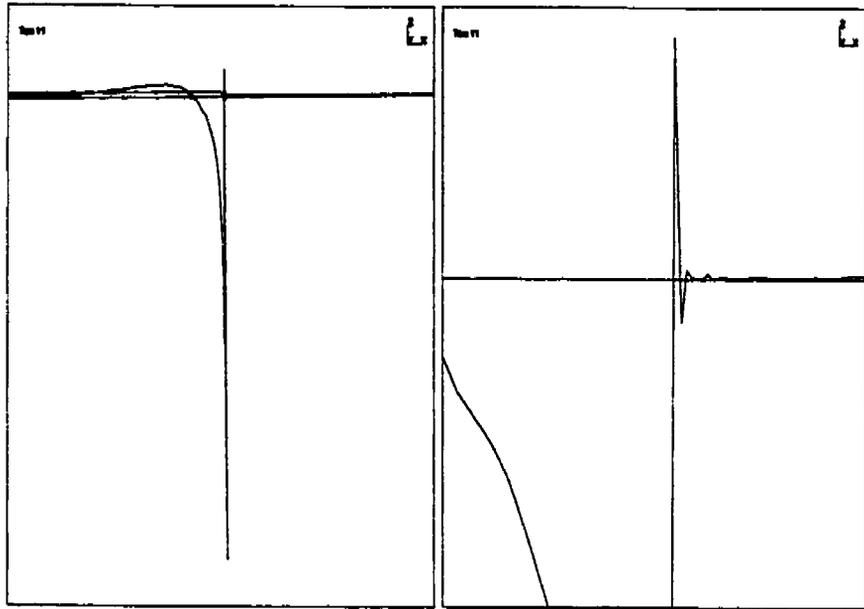


FIG. 5.36 - Coupe en $y = 1$ passant par la singularité pour τ_{11} sur ADAPT2 pour le problème de la contraction : Cas $\lambda = 0.0$

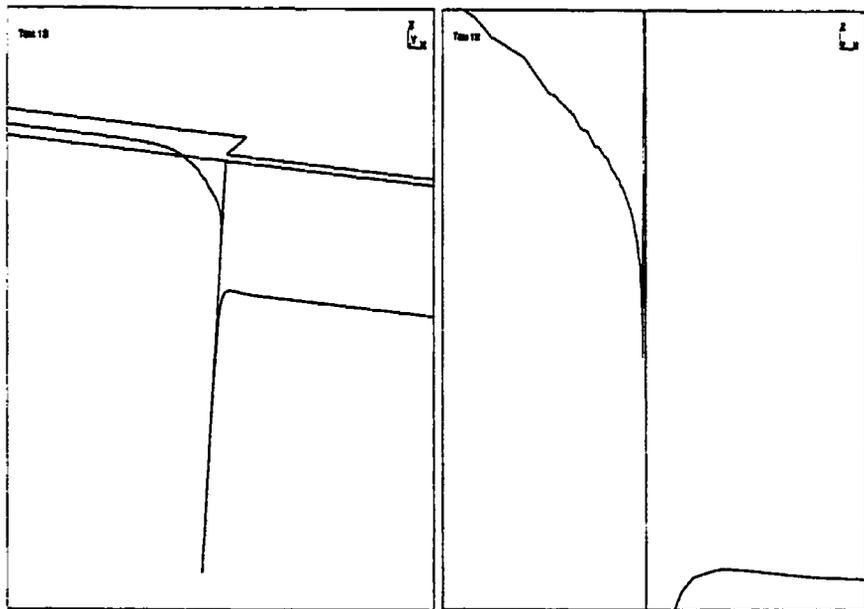


FIG. 5.37 - Coupe en $y = 1$ passant par la singularité pour τ_{12} sur ADAPT2 pour le problème de la contraction : Cas $\lambda = 0.0$

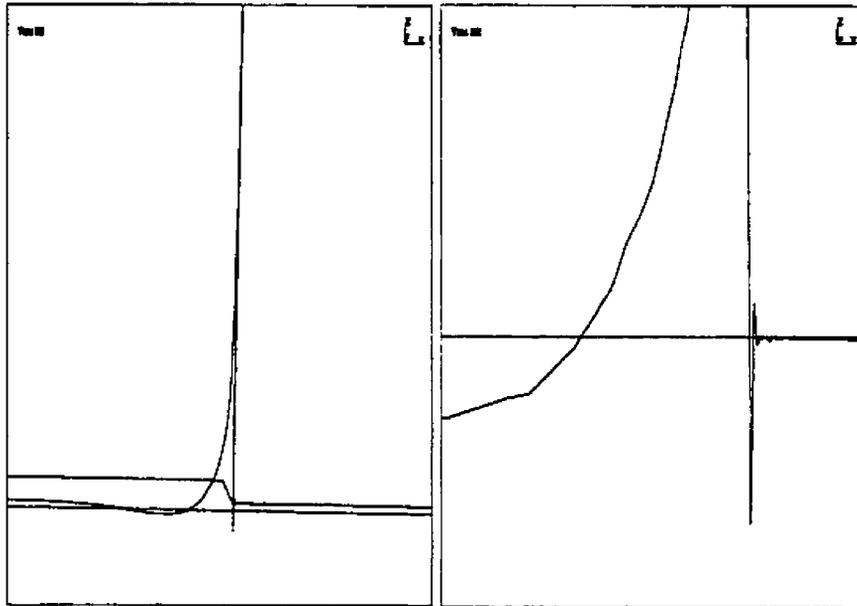


FIG. 5.38 – Coupe en $y = 1$ passant par la singularité pour τ_{22} sur ADAPT2 pour le problème de la contraction : Cas $\lambda = 0.0$

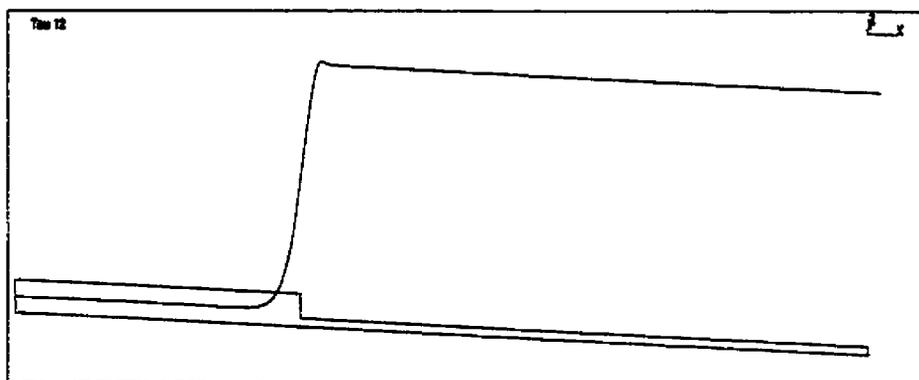


FIG. 5.39 – Coupe en $y = 0$ pour τ_{12} sur ADAPT2 pour le problème de la contraction : Cas $\lambda = 0.0$

5.5.12 Résolution du problème de la contraction : $\lambda = 1.0$

Dans le cas viscoélastique, nous avons effectué quatre adaptations et les maillages obtenus ADAPT1, ADAPT2, ADAPT3, et ADAPT4 sont illustrés à la figure 5.40. De plus, à la figure 5.41, un agrandissement du maillage final ADAPT4 près du coin (0,1). Les caractéristiques de ces maillages sont présentés au tableau 5.19 et les résultats d'adaptation au tableau 5.20.

Nous remarquons d'abord que l'adaptation de maillage s'est très bien comportée conservant la qualité moyenne des éléments qui est toujours supérieure à 0.85. Nous voyons aussi que le nombre de sommets a triplé durant la première adaptation et a, par la suite, fluctué en augmentant relativement peu. Nous avons donc une convergence rapide vers le maillage optimal. À noter que la longueur désirée fut $l_d = 100.0$.

On remarque que les solutions sont assez régulières et qu'une singularité est présente en (0,1) pour les variables τ_{11} , τ_{12} , τ_{22} , p . De plus, la singularité est transportée et diffusée tel que nous le voyons aux figures 5.45, 5.46 et 5.47 qui sont des coupes horizontales passant par la singularité. De plus, nous sommes en présence d'un problème de couche limite à la sortie de la contraction. On peut voir ce profil à la figure 5.42. À noter que, dans la figure 5.42, τ_{12} varie de -195.8 à 1052.2, mais nous avons tronqué à 300.0, question de bien voir la solution. D'autres solutions obtenus sont illustrés par les figures qui suivent.

	Maillages				
Qualité	ORIGINAL	ADAPT1	ADAPT2	ADAPT3	ADAPT4
Q_K moyenne	0.878633	0.874396	0.853883	0.859893	0.867426
Q_K minimum	0.383366	0.234769	0.0928123	0.106418	0.0358589
Q_K maximum	0.999948	0.999994	0.999997	0.999994	0.999994
Nombre d'éléments	4936	14737	13180	16889	21497
Nombre de sommets	2755	8059	7477	9537	11987

TAB. 5.19 – Caractéristiques des maillages générés pour le problème de la contraction :
Cas $\lambda = 1.0$.

	Maillages			
	ADAPT1	ADAPT2	ADAPT3	ADAPT4
Longueur calculée l_c	100.0	100.0	100.0	100.0
Nb raf. d'arêtes	12811	6130	10744	11945
Nb déraf. d'arêtes	8028	7067	8990	9719
Nb ret. d'arêtes	17828	13514	19441	20474
Nb dép. sommets	93178	77232	101200	129007
Dép. sommet Moyen	0.00218	0.00246	0.001912	0.001413
Dép. sommet Max.	0.05366	0.03510	0.032765	0.040203
Temps d'adaptation (sec.)	941.61	822.02	1068.92	1293.37
Mémoire utilisée (Octets)	24 901 656	22 464 744	28 754 808	36 490 776
Dimension Syst. Lin. (Nb. lignes)	69767	63743	81461	102927

TAB. 5.20 – Données relatives à l'adaptation pour le problème de la contraction : Cas
 $\lambda = 1.0$.

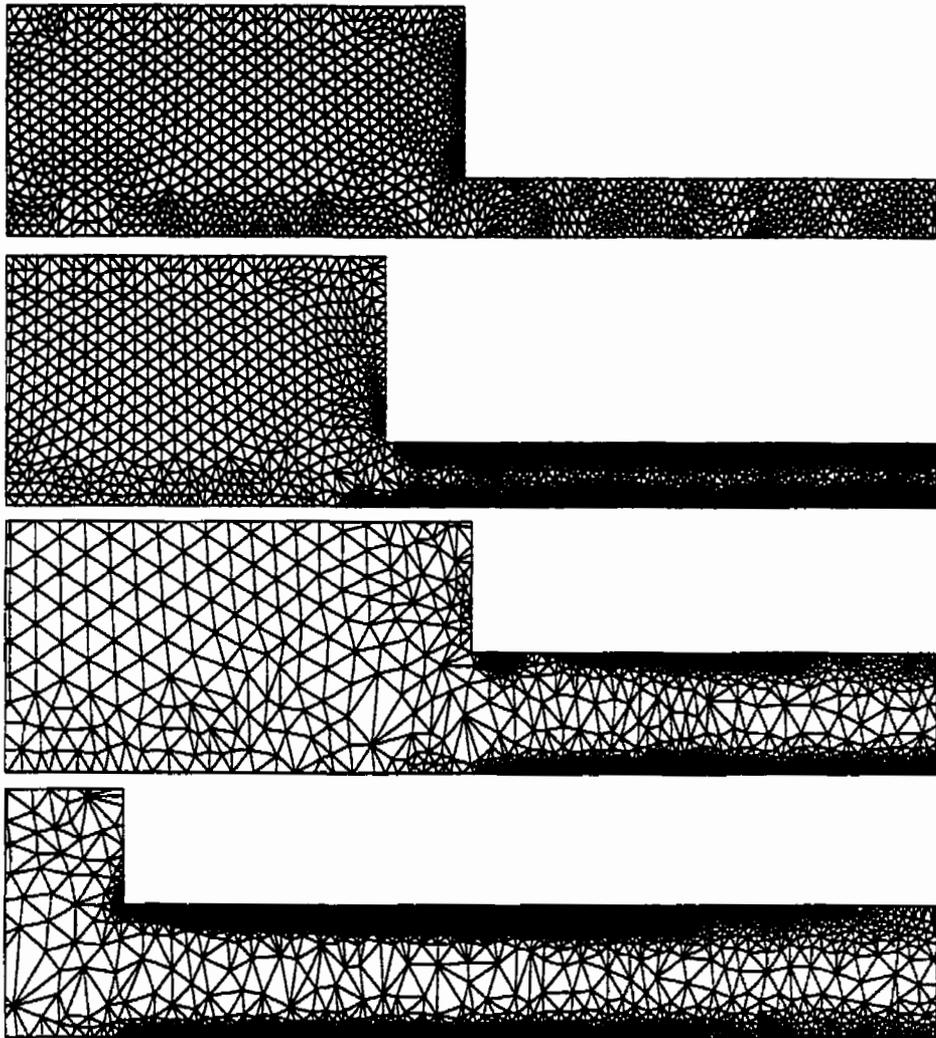


FIG. 5.40 – Agrandissements des maillages ADAPT1, ADAPT2, ADAPT3 et ADAPT4 (de haut en bas) générés pour le problème de la contraction : Cas $\lambda = 1.0$.

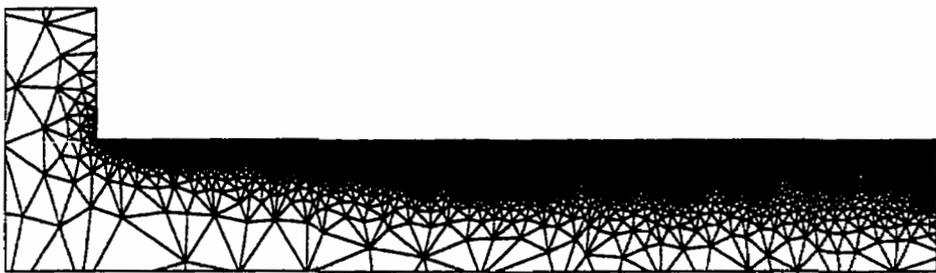


FIG. 5.41 – Agrandissement du maillage ADAPT4 près du coin (0,1) pour le problème de la contraction : Cas $\lambda = 1.0$

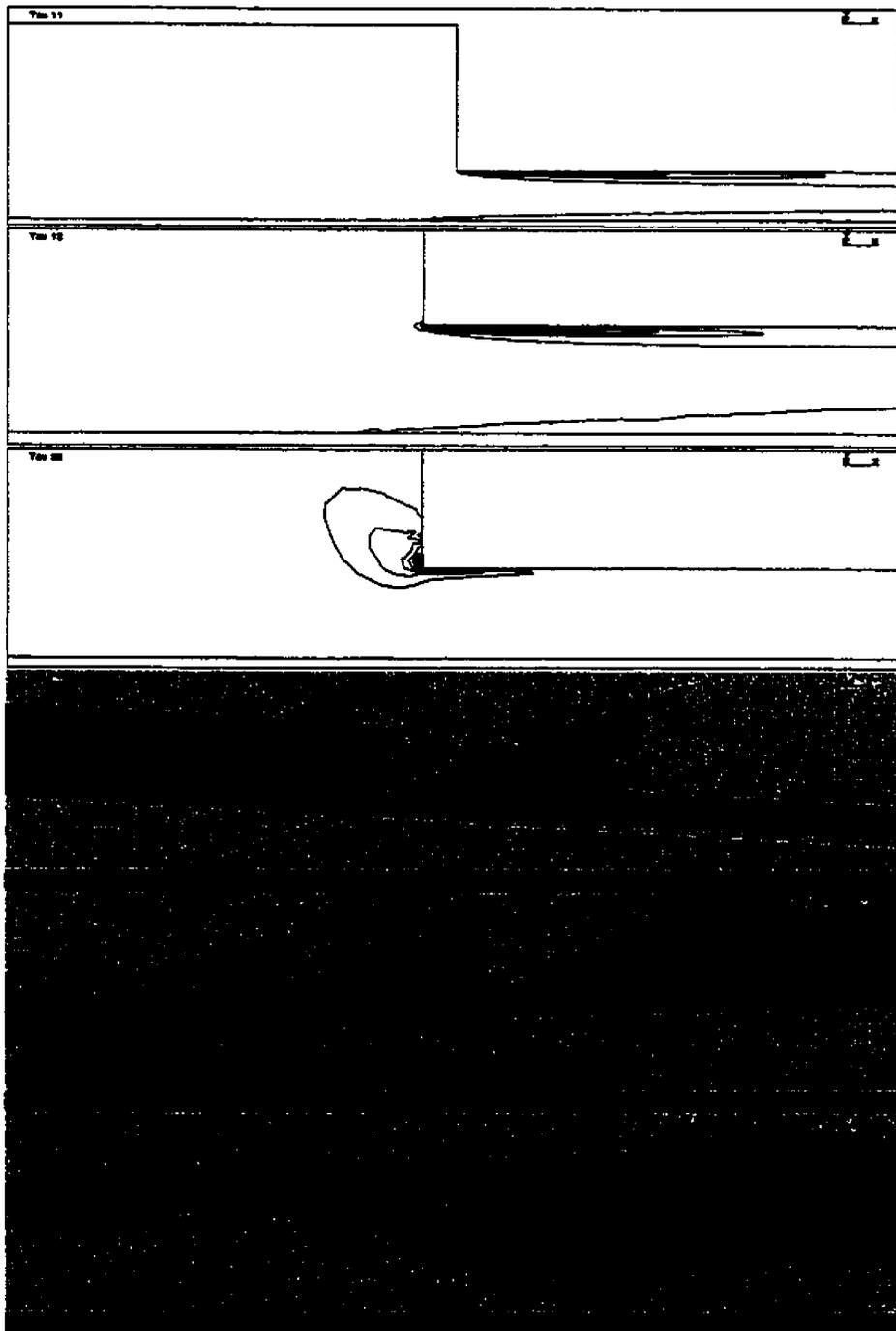


FIG. 5.42 – De haut en bas, les courbes de niveau des variables τ_{11} , τ_{12} et τ_{22} et par la suite les graphes des variables τ_{11} , τ_{12} et τ_{22} sur ADAPT4 pour le problème de la contraction : Cas $\lambda = 1.0$



FIG. 5.43 – Graphe de la variables p selon 2 vues différentes sur ADAPT4 pour le problème de la contraction : Cas $\lambda = 1.0$.



FIG. 5.44 – Graphe de la variable u sur ADAPT4 autour de l'obstacle pour le problème de la contraction : Cas $\lambda = 1.0$.

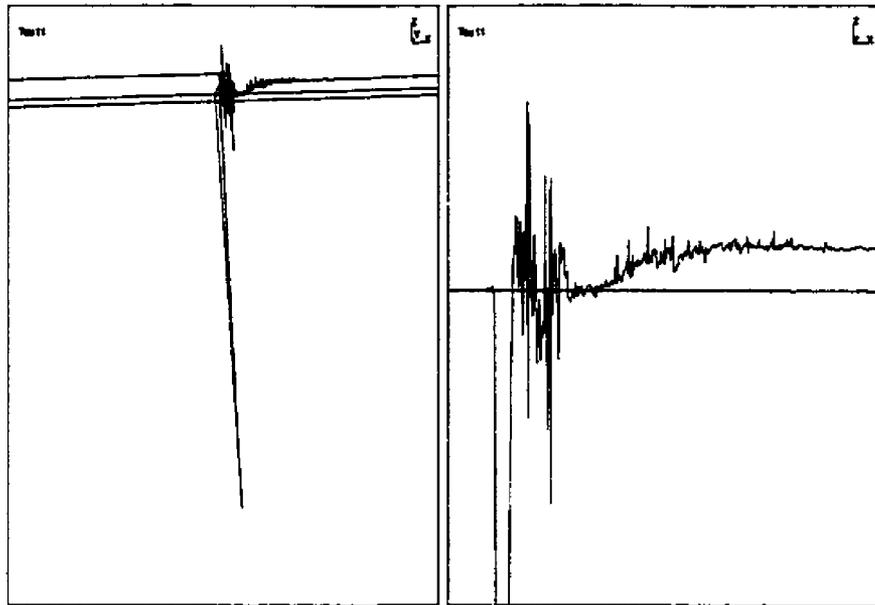


FIG. 5.45 – Coupe en $y = 1$ passant par la singularité pour τ_{11} sur ADAPT4 pour le problème de la contraction : Cas $\lambda = 1.0$

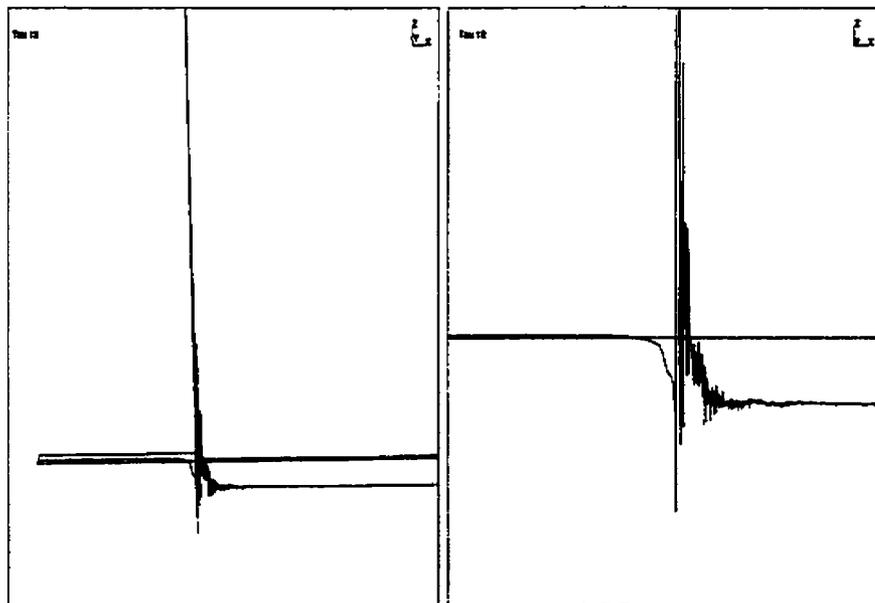


FIG. 5.46 – Coupe en $y = 1$ passant par la singularité pour τ_{12} sur ADAPT4 pour le problème de la contraction : Cas $\lambda = 1.0$

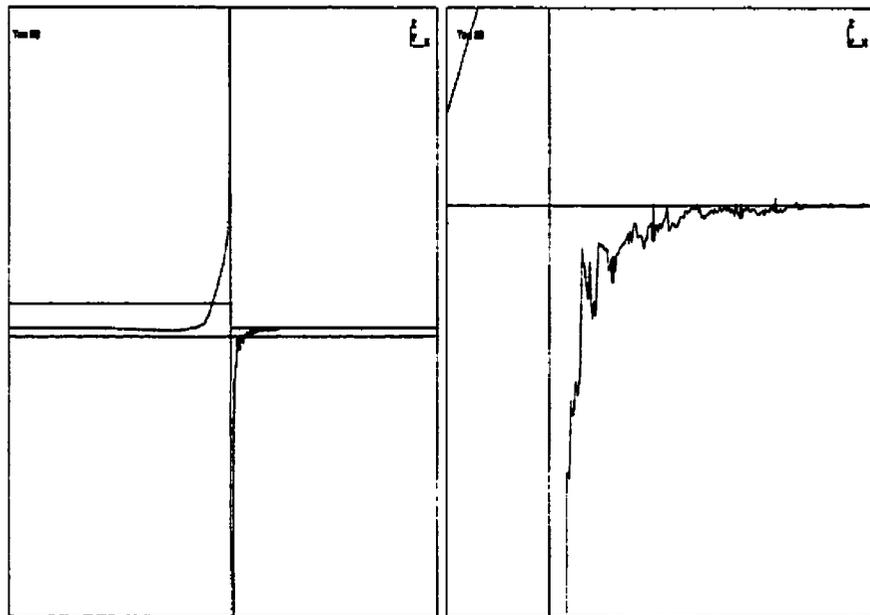


FIG. 5.47 - Coupe en $y = 1$ passant par la singularité pour τ_{22} sur ADAPT4 pour le problème de la contraction : Cas $\lambda = 1.0$

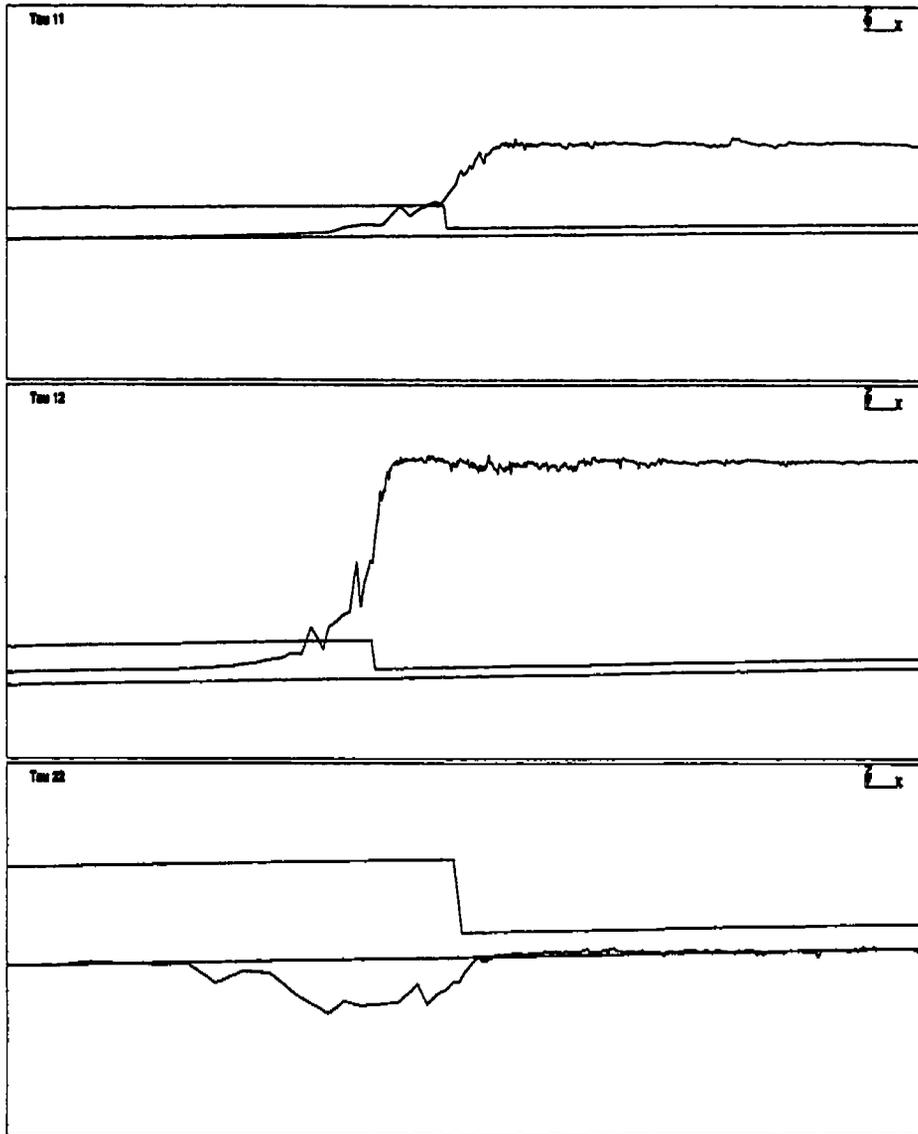


FIG. 5.48 – Coupe en $y = 0$ pour τ_{12} , τ_{12} et τ_{22} (De haut en bas) sur ADAPT4 pour le problème de la contraction : Cas $\lambda = 1.0$

Conclusion

Dans ce travail, nous avons développé un outil d'adaptation de maillages qui se révèle performant. On pourrait cependant discuter de la pertinence de poursuivre une telle recherche pour les problèmes bidimensionnels, alors même que la capacité en mémoire des ordinateurs croît très vite, permettant ainsi la résolution de problèmes de plus en plus gros. On peut, bien sûr, répondre que le passage à la dimension trois requiert encore de bonnes techniques de réduction des systèmes et qu'il est plus facile d'étudier ces techniques en dimension deux. De plus, nous prétendons que ces techniques sont indispensables même en dimension deux.

En effet, il faut bien voir que certains des problèmes résolus ici de façon découplée, n'auraient pas pu l'être de manière couplée, même sur des ordinateurs possédant quatre fois plus de mémoire et ce, à cause du grand nombre de variables physiques. En plus, il faut tenir compte des temps de calcul. Ainsi, pour résoudre un problème couplé, il faut augmenter graduellement la valeur de λ et, pour chaque λ fixé, résoudre le problème discret de façon itérative, ce qui exige plusieurs itérations que nous appellerons internes. Un test a été fait dans le code MEF++ pour résoudre le problème de la couche limite par une formulation couplée en utilisant l'algorithme de Newton dont les systèmes linéarisés sont résolus de nouveau par une autre méthode itérative. Nous avons eu divergence de la solution discrète après 40 incréments de 0.01 de la variable λ . Cette divergence était, en partie, due à la quantité, sans cesse croissante avec λ , du nombre d'itérations internes nécessaires. Ainsi, il est pratique courante de fixer un nombre maximum d'itérations internes par pas de λ et ce maximum fut atteint pour la valeur $\lambda = 0.35$. Ce maximum fut fixé à

10 et c'est seulement après 4 jours et plus de 200 systèmes résolus que nous avons observé la divergence de la solution. Par conséquent, nous estimons à un multiple de 10 000, le nombre de systèmes linéaires à résoudre pour obtenir la solution du système couplé, ce qui est déraisonnable. Évidemment, notre approche découplée ne nécessitait que la résolution de deux systèmes linéaires de tailles nettement inférieures, mais sans toutefois résoudre le problème original. Ceci montre clairement toute la difficulté des simulations numériques des écoulements de fluides viscoélastiques. D'autres travaux de recherche sont nécessaires pour d'obtenir une stratégie robuste et efficace de résolution de problèmes couplés.

C'est ici qu'intervient l'adaptation de maillages. Elle nous fournit une méthode relativement peu coûteuse pour diminuer le nombre de degrés de liberté tout en améliorant la précision. Dans le cas découplé, cette amélioration de la précision se traduit par une solution initiale plus précise pour le problème couplé et donc, une diminution considérable du nombre d'itérations internes qui exigent, en outre, la résolution de systèmes linéaires réduits.

Bien sûr, l'amélioration de la précision est intimement liée à la qualité de l'estimateur d'erreur *a posteriori* et à l'efficacité des algorithmes d'adaptation. Ce travail a montré que les algorithmes d'adaptation sont très performants. Bien que nous ayons testé qu'un seul estimateur d'erreur, les résultats sont très favorables. Il serait maintenant très intéressant de chercher à améliorer la précision en augmentant celle des estimateurs pour obtenir des solutions discrètes plus précises à moindre coût.

Bibliographie

- [1] I. Babuška, W. Rheinboldt, *A Posteriori Error Analysis of Finite Element Solutions for One-Dimensional Problems* Siam J. Numer. Anal., Vol 18 (3), 565-589, 1981.
- [2] R.E. Bank, R. K. Smith *A Posteriori Error Estimate Based on Hierarchical Bases* Siam J. Numer. Anal. Vol 30 (4), 921-935, 1993.
- [3] M. Bercovier, *Perturbation of a mixed variational problem, applications to mixed finite element methods*, R.A.I.R.O. Anal. Numer., 12, 211-236, 1978.
- [4] F. Brezzi, M. Fortin, *Mixed and Hybrid Finite Element Methods*, SCM 15, Springer Verlag, 1991.
- [5] P. G. Ciarlet *The finite element method for elliptic problem*, North Holland, 1978.
- [6] Duvaut, *Mécanique des milieux continus*, collection Mathématiques appliquées pour la maîtrise, Masson, Paris, 1990.
- [7] D. Esselaoui, *Méthodes numériques pour la simulation d'écoulements viscoélastiques*, Thèse, Université Laval, 1986.
- [8] M. Fortin, R. Guénette, R. Pierre, *Numerical Analysis of the modified EVSS method*, Comput. Meth. Appl. Mech. Engr., 143, 79-95, 1997.
- [9] W. G. Habashi, M. Fortin, D. Aït-Ali-Yahia, S. Boivin, Y. Bourgault, J. Dompierre, M. P. Robichaud, A. Tam, M. G. Vallet, *Anisotropic mesh optimization : towards a solver-independent and mesh-independent CFD*, VKI Lecture Series, 1996-06, 1996.
- [10] T.R.J. Hughes, J.E. Marsden, *A short course in fluid mechanics*, Mathematics Lecture Series, Publish or Perish Inc., Berkeley, 1976.
- [11] J. E. Marsden, A. J. Tromba, *Vector Calculus*, W.H. Freeman and Company, 1981.

- [12] P. A. Raviart, J. M. Thomas, *Introduction à l'analyse numérique des équations aux dérivées partielles*, 2^e éd. Masson, 1988.
- [13] M. Renardy, *Asymptotic structure of the stress field in flow past a cylinder at high Weissenberg number*, *Journal of Non-Newtonian Fluid Mechanics* 90 (1), 2000.
- [14] Y. Saad, *Iterative methods for sparse linear systems*, PWS Publishing Company, 1996.
- [15] R. B. Simpson, E. F. D'Azevedo, *On optimal triangular meshes for minimizing the gradient error*, *Numer. Math.* 59 (4), 1991.
- [16] R. Temam, *Navier-Stokes Equations*, North-Holland, 1984.
- [17] M.-G., Vallet, *Génération de maillages éléments finis anisotrope et adaptatifs*, Thèse, Université Paris 6, 1992.
- [18] R. Verfürth, *A posteriori error estimators for the Stokes equations*, *Numer. Math.*, 55 (1989), pp. 309–325.