

GINO PELLETIER

**ADAPTATION DE MAILLAGES ET FLEXION DES
PLAQUES**

Mémoire
présenté
à la Faculté des études supérieures
de l'Université Laval
pour l'obtention
du grade de maître ès sciences (M.Sc.)

**DÉPARTEMENT DE GÉNIE CIVIL
FACULTÉ DES SCIENCES ET GÉNIE
UNIVERSITÉ LAVAL**

FÉVRIER 1999



**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.

0-612-41985-1

Canada

Résumé

Nous présentons dans ce mémoire les résultats de nos travaux sur l'adaptation des plaques fléchies en utilisant comme critère d'adaptation l'erreur calculée à partir des dérivées secondes de la solution obtenue par éléments finis. Plusieurs cas ont été réalisés et analysés afin de démontrer l'efficacité de la méthode employée et la justesse des résultats obtenus. Nous avons programmé à l'aide du langage C++ et des bibliothèques "Diffpack" des classes pour la gestion des matériaux ainsi qu'un solveur éléments finis pour des éléments à trois nœuds tenant compte du cisaillement transverse mieux connu sous le nom de DST. La combinaison entre le solveur éléments finis et la bibliothèque d'adaptation de maillage "Libom" nous a permis d'obtenir un outil très efficace pour l'adaptation des plaques composites pour des domaines à deux dimensions.

Directeur de recherche :

M. Guy Gendron, Ph.D.

Co-directeur de recherche :

M. Michel Fortin, Ph.D.

Auteur :

M. Gino Pelletier, B.Sc.A.

Avant-propos

Les travaux de recherche présentés dans ce rapport ont été réalisés au sein du GIREF et du département de génie civil de l'université Laval.

Nous remercions tous les professeurs, collègues et autres personnes qui ont contribué, de près ou de loin, à la réalisation de ces travaux. Nous remercions également le GIREF pour l'accès aux équipements informatiques sans lesquels la réalisation de ce projet n'aurait pas été possible. Nous remercions également le Fonds FCAR pour le financement de cette recherche.

Finalement, j'aimerais remercier personnellement le professeur Guy Gendron, mon directeur, pour son support constant ainsi que de ses nombreuses suggestions (toujours pertinentes) tout au long de cette recherche. Aussi, j'aimerais remercier les professeurs M. Mario Fafard et M. Michel Fortin du GIREF pour avoir eu la patience et l'aimabilité d'avoir lu et commenté ce rapport.

Sainte-Foy, Québec
9 mai 1999

Gino Pelletier

Table des matières

Résumé	ii
Avant-propos	iii
Table des matières	iv
Liste des tableaux	vii
Liste des figures	ix
Symboles	xi
Notation	xii
1 Introduction	1
1.1 Généralités	1
1.2 Revue bibliographique	3
1.2.1 Programmation orientée-objet et méthode des éléments finis .	3
1.2.2 Adaptation de maillages	4
1.3 Contenu du mémoire	5
1.3.1 Objectifs et apports du mémoire	5
1.3.2 Contenu des chapitres	6

2	Description de l'élément DST	8
2.1	Description de l'élément de membrane CST	8
2.1.1	Relations cinématiques	8
2.1.2	Lois de comportement	9
2.1.3	Forme variationnelle faible	10
2.1.4	Formulation de l'élément CST	11
2.2	Description de l'élément de flexion DST [7]	13
2.2.1	Relations déformations-déplacements	13
2.2.2	Relation contraintes-déformations et efforts résultants	16
2.2.3	Formulation détaillée de l'élément DST	17
2.3	Assemblage	31
2.3.1	Définition du repère local	32
2.3.2	Expression de W_{int}^e dans le repère local	33
2.3.3	Rigidité fictive pour θ_z	34
2.3.4	Expression de W^e dans le repère global	35
3	Adaptation de maillages	37
3.1	Description de l'estimateur d'erreur utilisé	37
3.1.1	Évaluation de l'erreur d'interpolation	39
3.1.2	Évaluation du Hessien	41
3.1.3	Évaluation de l'erreur sur une arête	43
3.1.4	Remarques	44
3.2	Stratégie d'adaptation de maillage	45
4	Résultats de l'adaptation de maillage	47
4.1	Cas 1 : Poutre en porte-à-faux cisailée	49
4.1.1	Résultats de l'adaptation de maillage	50

4.2	Cas 2 : Poutre fléchie en porte-à-faux	53
4.2.1	Résultats de l'adaptation de maillage	55
4.2.2	Discussion	61
4.3	Cas 3 : Poutre fléchie simplement supportée	63
4.3.1	Résultats de l'adaptation de maillage	63
4.3.2	Discussion	66
4.4	Cas 4 : Plaque carrée orthotrope	70
4.4.1	Géométrie et propriétés du modèle	70
4.4.2	Résultats de l'adaptation de maillage	71
4.4.3	Discussion	75
4.5	Cas 5 : Plaque trouée orthotrope	79
4.5.1	Géométrie et propriétés du modèle	79
4.5.2	Résultats de l'adaptation de maillage	80
4.5.3	Discussion	85
5	Conclusion	88
5.1	Récapitulation et conclusion	88
5.2	Recommandations et perfectionnements	89
	Bibliographie	91
A	Lois de comportement des plaques	94
A.1	Matériau isotrope	95
A.1.1	Élasticité tridimensionnelle	95
A.1.2	Élasticité 2D (hypothèse des contraintes planes)	95
A.1.3	Élasticité 2D (hypothèse des déformations planes)	97
A.2	Matériau orthotrope	98
A.2.1	Matériau orthotrope dans 3 directions	99

A.2.2	Matériau orthotrope dans le plan LT et isotrope dans le plan T_z	100
A.2.3	Rotations des contraintes et des déformations	101
A.3	Matériau composite	102
A.3.1	Calcul de la matrice $[H_c]$	105
B	Gestion des matériaux	110
B.1	Classe Matériaux et MatériauxElastique	111
B.2	Classe Isotrope	112
B.3	Classe Orthotrope	113
B.4	Classe Pli	114
B.5	Classe Composite	116

Liste des tableaux

4.1	Calcul d'erreur sur les arêtes pour le cas 1	51
4.2	Résultats vs longueur optimale dans la métrique pour le cas 1	53
4.3	Calcul d'erreur sur les arêtes pour le cas 2 (sans restriction)	55
4.4	Calcul d'erreur sur les arêtes pour le cas 2 (limite = 3.33 mm)	61
4.5	Calcul d'erreur sur les arêtes pour le cas 2 (limite = 5.0 mm)	61
4.6	Calcul d'erreur sur les arêtes pour le cas 2 (limite = 10.0 mm)	62
4.7	Calcul d'erreur sur les arêtes pour le cas 3 (sans restriction)	64
4.8	Calcul d'erreur sur les arêtes pour le cas 3 (limite = 5.0 mm)	66
4.9	Calcul d'erreur sur les arêtes pour le cas 3 (limite = 10.0 mm)	69
4.10	Calcul d'erreur sur les arêtes pour le cas 4	75
4.11	Résultats vs longueur optimale dans la métrique pour le cas 4	75
4.12	Statistiques sur le calcul de la contrainte σ_x pour le cas 4	78
4.13	Statistiques sur le calcul des contraintes pour le cas 5	83
4.14	Résultats vs Longueur optimale dans la métrique pour le cas 5	83

Liste des figures

2.1	Élément de membrane	11
2.2	Élément triangulaire à 9 degrés de liberté	14
2.3	Déformations e , $z\kappa$ et γ	16
2.4	Efforts résultants sur un élément de plaque	17
2.5	Élément réel et système de coordonnées local sur un côté	20
2.6	Variation des rotations β_s et β_n sur les côtés	23
2.7	Construction de l'élément DST18	31
3.1	Description des domaines et de leur frontière	42
3.2	Description du processus d'adaptation	46
4.1	Description du modèle pour le cas 1	49
4.2	Maillages obtenus au cours du processus d'adaptation	50
4.3	Erreur moyenne vs nombre de nœuds pour le cas 1	52
4.4	Maillages obtenus pour différentes longueurs optimales dans la métrique	53
4.5	Courbe de convergence pour le cas 1	54
4.6	Description du modèle pour le cas 2	54
4.7	Maillages obtenus au cours du processus d'adaptation (sans restriction)	56
4.8	Maillages obtenus au cours de l'adaptation (limite = 3.33 mm)	58
4.9	Maillages obtenus au cours de l'adaptation (limite = 5.0 mm)	59
4.10	Maillages obtenus au cours de l'adaptation (limite = 10.0 mm)	60

4.11	Erreur vs nombre de nœuds (Cas 2)	62
4.12	Description du modèle pour le cas 3	63
4.13	Maillages obtenus au cours du processus d'adaptation (sans restriction)	65
4.14	Maillages obtenus au cours de l'adaptation (limite = 5.0 mm)	67
4.15	Maillages obtenus au cours de l'adaptation (limite = 10.0 mm)	68
4.16	Erreur vs nombre de nœuds (Cas 3)	69
4.17	Description du modèle pour le cas 4	70
4.18	Maillages obtenus au cours du processus d'adaptation	72
4.19	Graphiques des dérivées secondes pour le cas 4	73
4.20	Maillages obtenus pour différente longueur optimale dans la métrique	74
4.21	Isocontours pour les composantes θ_x , θ_y et w du déplacement (Cas 4)	76
4.22	Erreur vs nombre de nœuds (Cas 4)	77
4.23	Convergence maillages uniformes vs convergence maillages adaptés pour le cas 4	78
4.24	Relation entre l'erreur sur les arêtes et l'écart entre les contraintes . .	79
4.25	Description du modèle pour le cas 5	80
4.26	Maillages obtenus au cours du processus d'adaptation	81
4.27	Maillages obtenus pour différente longueur optimale dans la métrique	84
4.28	Isocontours pour les composantes θ_x , θ_y et w du déplacement (Cas 5)	85
4.29	Erreur vs nombre de nœuds (Cas 5)	86
4.30	Convergence maillages uniformes vs convergence maillages adaptés pour le cas 5	87
A.1	Système d'axe local	98
A.2	Système d'axe local et global	101
A.3	Variables utilisées pour les calculs	103
B.1	Hierarchie des classes pour la gestion des matériaux	111

Symboles

x (caractère gras)	vecteur ou matrice
$\{x\}$	vecteur colonne
$\langle x \rangle$	vecteur ligne
$[x]$	matrice
$[x]^T$	transposé d'une matrice
$[x]^{-1}$	inverse d'une matrice
$\det[x]$	déterminant d'une matrice
$\ x\ $	norme euclidienne (L_2)
\sum	sommation
\int	intégrale
cos	cosinus
sin	sinus
tan	tangente
\wedge	produit vectoriel

Notation

Chapitre 2

$\{\sigma\}$	vecteur des contraintes
$\{\epsilon\}$	vecteur des déformations
$[H]$	matrice de comportement
$\{\tau\}$	vecteur des contraintes de cisaillement transverse
$[H_\tau]$	matrice de comportement en cisaillement transverse
$\{\gamma\}$	vecteur des déformations de cisaillement transverse
β_x	rotation dans le plan X-Z
β_y	rotation dans le plan Y-Z
$\{\kappa\}$	vecteur des déformations de flexion (courbures)
$\{e\}$	vecteur des déformations de membrane
$\{\sigma_L\}$	vecteur des contraintes dans la direction longitudinale
$[H_L]$	matrice de comportement dans le système d'axe local
$\{\epsilon_L\}$	vecteur des déformations dans le système d'axe local
$\{\tau_L\}$	vecteur des contraintes de cisaillement transverse dans le système d'axe local
$[H_{\tau L}]$	vecteur de comportement en cisaillement transverse dans le système d'axe local
$\{\gamma_L\}$	vecteur des déformations de cisaillement transverse dans le système d'axe local
$[T_1]$	matrice de rotation des contraintes
$[T_2]$	matrice de rotation des contraintes
$\{N\}$	vecteur des efforts normaux
$\{M\}$	vecteur des moments fléchissants
$\{T\}$	vecteur des efforts tranchants

$[H_m]$	matrice des déformations de membrane
$[H_{mf}]$	matrice de couplage entre les effets de membrane et flexion
$[H_f]$	matrice des déformations de flexion
$[H_c]$	matrice des déformations de cisaillement

Chapitre 3

$[E], [\epsilon]$	tenseur des déformations de Green-Lagrange
$[F]$	tenseur gradient de déformations
$[\sigma]$	tenseur des contraintes
$W_{\text{int}}, W_{\text{ext}}$	travail virtuel interne et externe
dV	élément différentiel de volume
$[B]$	matrice reliant les déformations aux variables nodales
$\{u\}$	variables nodales
J	déterminant de la matrice jacobienne
$[J]$	matrice jacobienne
$[j]$	matrice jacobienne inverse
$[N_i]$	fonctions d'interpolation i
$[k^e]$	matrice de rigidité élémentaire
u, v, w	déplacements selon x, y et z
$\{\tilde{\gamma}\}$	vecteur des déformations de cisaillement transverse de substitution
U_c^e	énergie interne de déformation de cisaillement transverse
C_k, S_k	cosinus et sinus directeur
$\{u_n\}$	vecteur des variables nodales
$\{\alpha_k\}$	vecteur des variables non-nodales
$[B_f]$	matrice reliant les variables nodales et non-nodales aux courbures
$\{a_1\}, \{a_2\}$	vecteur de base covariants
$\{n\}$	vecteur normal à la surface de l'élément
$\{t_1\}, \{t_2\}$	vecteurs tangent à l'élément
$[k_m]$	matrice de rigidité de membrane
$[k_{mf}]$	matrice de couplage entre la rigidité de membrane et flexion
$[k_f]$	matrice de rigidité de flexion
$[k_{\theta z}]$	matrice de rigidité fictive de rotation

[Q] matrice de transformation orthonormée
[T] matrice de rotation

Chapitre 1

Introduction

1.1 Généralités

Pour des problèmes relativement simples, l'ingénieur-concepteur peut encore compter sur les calculs manuels. Par contre, pour des problèmes plus complexes, l'ordinateur devient un outil indispensable. Pour les problèmes d'analyse des contraintes, la méthode des éléments finis est une méthode très populaire. Celle-ci permet de résoudre des problèmes qui ne pourraient être résolus autrement. Par conséquent, on veut de plus en plus que tous les ingénieurs soient en mesure de faire des analyses par la méthode des éléments finis.

La taille des problèmes à résoudre en mécanique des solides étant de plus en plus importante, la nécessité de diminuer les temps de calcul est rapidement devenue, pour les utilisateurs de logiciels de calcul par éléments finis, un besoin très important. Bien sûr, l'amélioration et l'optimisation des méthodes pour la résolution des systèmes linéaires d'envergure a grandement aidé à faire diminuer les temps de calcul mais, la résolution des problèmes demeure encore extrêmement coûteuse en temps et par conséquent en argent.

Le temps que l'ingénieur consacre à l'élaboration et à la préparation du maillage est aussi un facteur non négligeable dans le temps et le coût total associés à

une analyse par éléments finis. Un besoin est alors apparu pour le développement d'outils qui aideront l'utilisateur dans la définition des maillages pour la résolution par éléments finis. Jusqu'à un passé très récent, et encore très largement aujourd'hui, c'est l'utilisateur qui définit de façon seule et intuitive le maillage à utiliser. En laissant à ce dernier l'entière liberté du maillage à utiliser, il arrive parfois que celui-ci soit trop dense dans certaines zones ou plus grave, trop grossier dans d'autres. Une conséquence directe de cet état de fait est que la précision de la solution obtenue est très variable. Il est aussi très difficile pour un utilisateur sans expérience et même pour un utilisateur expérimenté de définir un maillage qui sera à la fois assez dense pour obtenir une solution précise et assez grossier pour ne pas obtenir des temps de calculs trop longs. Le but est donc d'obtenir un maillage qui pourra le mieux possible représenter la solution voulue tout en obtenant des temps de calculs raisonnables et ceci de façon automatique.

Nous proposons, pour atteindre cet objectif, de progressivement modifier un maillage afin de répartir uniformément l'erreur. Pour cela, le maillage devra être plus ou moins dense selon que la solution recherchée varie fortement ou non.

Afin d'obtenir ce maillage *adapté*, l'erreur devra être connue en tout point sur un maillage de départ afin de le modifier de façon adéquate. Les tâches à effectuer consisteront donc à déplacer ou à enlever des nœuds et à ajouter ou à enlever des éléments selon que l'erreur est grande ou petite pour permettre une uniformisation de l'erreur sur le domaine discrétisé. Ainsi, pour un niveau d'erreur donné, le maillage devra être fin dans les régions de fortes variations de la solution et plus grossier dans les zones de faibles variations de la solution.

Le maillage optimisé sera obtenu à la suite d'un processus d'adaptation. Sommairement, ce processus consistera à obtenir une solution par éléments finis à l'aide d'un logiciel de calcul pour ensuite estimer l'erreur sur ce maillage. Ayant en main un estimé de l'erreur sur un maillage de départ, la prochaine étape est celle d'adaptation. C'est au cours de la phase d'adaptation que le maillage de départ sera modifié. À partir de ce maillage modifié, une nouvelle solution est calculée. À partir de cette solution, l'erreur sur le maillage est estimée de nouveau pour ensuite recommencer la phase d'adaptation et ainsi de suite. Ce processus itératif devrait normalement converger vers un maillage *adapté optimisé* qui rejoint l'objectif mentionné plus haut

de répartir uniformément l'erreur.

1.2 Revue bibliographique

1.2.1 Programmation orientée-objet et méthode des éléments finis

La méthode des éléments finis a connu un essor considérable au cours des 15 dernières années. Avec la progression constante de la puissance des ordinateurs et le faible coût de ceux-ci, la méthode des éléments finis est devenue à la portée de tous (d'où l'importance d'avoir à sa disposition des outils pour le calcul de l'erreur et d'adaptation). En parallèle avec la progression des ordinateurs, plusieurs personnes ont cherché à améliorer les techniques de programmation. La programmation **orienté-objet** a alors fait son apparition.

La programmation orienté-objet est maintenant devenu très populaire en éléments finis. Au cours des dernières années, la quantité d'articles sur ce sujet a considérablement augmenté [14, 21]. C'est à cause de sa grande capacité à résoudre les problèmes complexes que la programmation orientée-objet est devenue aussi populaire. En effet, un des grands désavantages de l'utilisation des langages procéduraux comme le C ou le Fortran est le fait que la complexification du code augmente très rapidement avec l'augmentation de la taille de celui-ci. L'utilisation d'un langage orienté-objet comme le C++ ou le Java permet une meilleure structuration du code en forçant le programmeur à définir de façon précise les tâches à effectuer. Chacune de ces tâches sera la responsabilité d'une entité particulière dans le code, soit l'**objet**.

Le code adoptant une structure orientée-objet est composé d'un nombre déterminé d'objets qui échangent de l'information entre eux afin de résoudre le problème. Cette manière de procéder permet, en plus d'une simplification du code, une réutilisation éventuelle des parties du code. Pour plus d'information au sujet de la programmation orienté-objet, le lecteur peut se référer à [15].

À cause de ces avantages, nous avons décidé d'utiliser le langage C++ pour

réaliser notre programme de calcul par éléments finis. Plusieurs codes éléments finis écrits en C++ sont disponibles dans internet. Nous avons choisi d'utiliser les bibliothèques "Diffpack" développées à l'université d'Oslo par les départements de mathématique et d'informatique en collaboration avec *SINTEF Applied Mathematics*. Ces bibliothèques offrent de nombreuses possibilités. Par exemple, elles permettent la résolution des systèmes d'équations linéaires, la génération automatique de documents HTML ou L^AT_EX, le prototypage rapide d'un résolveur éléments finis, etc. Les bibliothèques "Diffpack" sont également munies d'outils de post-traitement et de pré-traitement qui facilitent la création des maillages et la visualisation des résultats. Ces possibilités et plusieurs autres sont décrites aux références [9, 10, 11].

1.2.2 Adaptation de maillages

Plusieurs procédures d'adaptation de maillage ont été développées au cours des dernières années avec toujours comme objectif l'obtention d'un maillage *optimisé*. Par maillage optimisé, on parle ici d'un maillage qui permet d'obtenir une erreur uniforme sur tout le domaine. C'est-à-dire que l'erreur calculée en différents endroits du domaine discrétisé devra être la même. Pour cela, le maillage devra être plus ou moins dense selon que la solution recherchée varie fortement ou faiblement.

Les méthodes obtenues jusqu'à maintenant diffèrent souvent que par la façon dont l'erreur est estimée. L'estimateur d'erreur le plus rencontré dans la littérature est celui développé par Zienkiewicz et Zhu [31]. Celui-ci estime l'erreur de discrétisation (locale et globale) d'une solution éléments finis en comparant les contraintes, discontinues entre éléments, avec des contraintes *lissées* [32, 33]. D'autres auteurs ont développé des estimateurs d'erreur basés sur le même principe. Tessler *et al.* [29] ont amélioré la technique de post-traitement des contraintes ce qui améliore l'estimation de l'erreur.

Le grand désavantage dans l'utilisation des estimateurs d'erreur de type Zienkiewicz-Zhu est le fait que ce dernier est directement reliés à la physique du problème. En effet, cet estimateur n'est valide que pour l'analyse des contraintes. Également, ce type d'estimateur nous permet de n'obtenir que des informations sur la dimension des éléments. Des informations comme la forme des éléments ou la direction dans

laquelle ceux-ci devraient être orientés ne sont pas disponibles avec ces estimateurs.

À partir d'un estimateur d'erreur, un maillage doit être généré ou modifié de façon à équidistribuer l'erreur jusqu'à un niveau voulu. Une stratégie d'adaptation de maillage consiste donc à enrichir le maillage ou le degré d'interpolation dans certaines zones en regard d'un estimateur d'erreur de façon à améliorer la solution numérique. Les stratégies d'adaptation de maillage peuvent être classées selon cinq catégories [13] :

- mouvement de nœuds (*r-method*) ;
- raffinement/déraffinement de maillage (*h-method*) ;
- enrichissement du degré d'interpolation (*p-method*) ;
- remaillage ;
- méthodes combinées.

Toutes ces méthodes ont été largement discutées dans la littérature. Le lecteur peut consulter Habashi *et al.* [13] pour avoir plus d'information sur ce sujet.

1.3 Contenu du mémoire

1.3.1 Objectifs et apports du mémoire

Les travaux présentés dans ce rapport ont comme objectif principale de réaliser des modèles éléments finis de plaques fléchies avec adaptation automatique de maillage. Étant donné que la *bibliothèque d'adaptation de maillage* "Libom" mise à notre disposition ne permet que l'analyse des domaines en deux dimensions, seules des plaques 2D ont été étudiées. De plus, un autre objectif de notre travail est d'étendre les fonctionnalités de la bibliothèque "Diffpack" pour permettre l'étude de plaques laminées.

En ayant comme objectif principal de réaliser des modèles éléments finis de plaques fléchies avec adaptation de maillage, plusieurs décisions ont dû être prises.

Premièrement, il a été choisi d'utiliser les bibliothèques "Diffpack" pour concevoir un logiciel de calcul par éléments finis pour l'analyse des plaques fléchies. Ce logiciel, conçu à l'aide du langage C++ et utilisant la philosophie de programmation orientée-objet, permet l'analyse des plaques discrétisées à l'aide d'éléments triangulaires à trois nœuds tenant compte du cisaillement transverse (DST18).

Pour tenir compte du fait que les plaques pourront être formées d'un matériau composite, des classes pour la gestion des matériaux ont été développées. Ces classes permettent le calcul des lois de comportement des plaques pour des matériaux isotropes, orthotropes ou stratifiés.

Ayant en main le logiciel d'analyse par éléments finis, il nous a été possible de développer une interface avec le logiciel d'adaptation de maillage "Libom" décrit aux références [12] et [13]. On a pu ainsi réaliser des analyses de modèles éléments finis avec adaptation automatique de maillage.

Les apports du mémoire sont donc :

- Conception et programmation de classes en C++ pour la gestion des matériaux et plus particulièrement des stratifiés ;
- Programmation d'un élément triangulaire à trois nœuds tenant compte du cisaillement transverse pour l'analyse des plaques fléchies orientées dans l'espace à trois dimensions ;
- Création d'une interface entre les logiciels conçus et le logiciel d'adaptation de maillage "Libom" ;
- Analyse de modèles éléments finis avec adaptation automatique de maillage et validation de la bibliothèque d'adaptation de maillage "Libom".

1.3.2 Contenu des chapitres

Le mémoire est divisée en six chapitres. Après une introduction générale et une revue de la bibliographie qui font l'objet du présent chapitre, nous présentons au second chapitre, la description de l'élément utilisé dans les modèles éléments finis (DST18). Cet élément est composé de l'élément DST [4] pour la partie flexion et de

l'élément CST pour la partie membrane. Le tout est assemblé au niveau global pour former un élément ayant six degrés de liberté par nœud pour un total de 18 degrés de liberté, d'où le nom de DST18.

La description du processus d'adaptation de maillage fait l'objet du troisième chapitre. Dans ce chapitre, nous présentons les bases théoriques pour l'estimation de l'erreur sur le maillage. Nous faisons également une brève description des stratégies utilisées afin de modifier le maillage de départ pour obtenir un maillage optimisé.

Dans le quatrième chapitre, nous présentons plusieurs résultats des analyses qui ont été faites à l'aide des outils développés dans le cadre de nos travaux. Plusieurs cas ont été étudiés afin de démontrer que l'adaptation de maillage dans un contexte de plaques fléchies utilisant les méthodes présentés au chapitre 3 donne de bons résultats. Tout au long de la présentation des résultats, plusieurs détails sont donnés sur la validité des résultats obtenus.

Finalement, nous présentons, dans le dernier chapitre, les conclusions générales et soulevons quelques questions qui devraient faire l'objet de recherches futures.

Chapitre 2

Description de l'élément DST

L'élément utilisé pour l'adaptation de maillage est un élément de type facette plane. Cet élément peut être utilisé pour représenter la géométrie d'une coque de forme quelconque par un ensemble de facettes planes triangulaires définies par trois nœuds. Cette approche est très utilisée en pratique. Cet élément est obtenu en combinant des éléments de membrane et de flexion. Le couplage des effets de membrane et de flexion est réalisé au niveau de l'assemblage par transformation des variables nodales locales vers un repère global.

Dans les sections qui suivent, nous présentons en détail l'élément appelé DST18, où la partie membrane est représentée par l'élément linéaire CST et la partie flexion par l'élément DST. La section 2.1 sera consacrée à la description de l'élément de membrane CST. La section 2.2 sera, quant à elle, consacrée à la description de la partie flexion à l'aide de l'élément DST. Finalement, l'assemblage de la partie membrane et de la partie flexion sera discuté à la section 2.3. Également dans cette section, la transformation du niveau local vers le niveau global sera discutée.

2.1 Description de l'élément de membrane CST

2.1.1 Relations cinématiques

La théorie linéaire classique de l'élasticité est basée sur les hypothèses suivantes :

- Les gradients des déformations sont supposés petits. Ceci implique que les produits et les carrés des premières dérivées sont négligeables.

$$u_{,x^0}, u_{,y^0}, u_{,z^0} \ll 1$$

$$(u_{,x^0})^2, (u_{,y^0})^2, (u_{,x^0}u_{,y^0}) \ll u_{,x^0}, u_{,y^0}$$

- Les déplacements u , v et w sont petits par rapport aux dimensions du solide. Nous pouvons donc confondre les configurations déformée et non déformée.

$$\begin{aligned} dV &\cong dV^0 \\ \frac{\partial}{\partial x} &\cong \frac{\partial}{\partial x^0} \end{aligned}$$

Pour une mesure des déformations du type Green-Lagrange, les relations cinématiques entre les déplacements (u, v) et les déformations $(\epsilon_{xx}, \epsilon_{yy}, \dots)$ sont données par l'équation suivante :

$$[\epsilon] = [E] = \frac{1}{2} ([F]^T [F] - [I]) \quad (2.1)$$

avec dans le cas à deux dimensions

$$[F] = \begin{bmatrix} 1 + u_{,x} & u_{,y} \\ v_{,x} & 1 + v_{,y} \end{bmatrix} \quad (2.2)$$

Le tenseur de Green-Lagrange $[E]$ est composé d'une partie linéaire et d'une partie quadratique. Puisque la partie quadratique est considérée comme négligeable par rapport à celle linéaire (gradients des déformations petits), nous obtenons donc :

$$[\epsilon] = \begin{bmatrix} \epsilon_{xx} & \gamma_{xy} \\ \gamma_{zy} & \epsilon_{yy} \end{bmatrix} = \begin{bmatrix} u_{,x} & (u_{,y} + v_{,x}) \\ (u_{,y} + v_{,x}) & v_{,y} \end{bmatrix} \quad (2.3)$$

2.1.2 Lois de comportement

Pour le cas de l'élasticité linéaire à deux dimensions, les différentes lois de comportement sont données à l'annexe A. La section A.1.2 (page 95) nous a donné la loi de comportement pour l'élasticité linéaire isotrope avec hypothèses des contraintes planes. La section A.1.3 (page 97) nous a donné la loi de comportement pour l'élasticité linéaire isotrope avec hypothèses des déformations planes. Finalement, la section A.3 (page 102) nous donne la loi de comportement relative au laminé.

2.1.3 Forme variationnelle faible

Les équations d'équilibre sur le volume V sont données par les équations qui suivent :

$$\begin{aligned}\sigma_{xx,x} + \sigma_{xy,y} + f_x &= 0 \\ \sigma_{xy,x} + \sigma_{yy,y} + f_y &= 0\end{aligned}\tag{2.4}$$

Les équations d'équilibre peuvent aussi s'exprimer sous la forme suivante :

$$\text{div}([\sigma]) + \{f_v\} = \{0\}\tag{2.5}$$

À partir de l'équation 2.5, il est possible d'obtenir la forme intégrale forte à l'aide de la méthode des résidus pondérés :

$$W = - \int_V \psi (\text{div}([\sigma]) + \{f_v\}) dV = 0\tag{2.6}$$

En utilisant une pondération de Galerkin et après une intégration par parties, on obtient la forme variationnelle faible suivante :

$$W = W_{\text{int}} - W_{\text{ext}} = 0\tag{2.7}$$

avec

$$W_{\text{int}} = \int_V (\delta u_{,x} \sigma_{xx} + \delta v_{,y} \sigma_{yy} + (\delta u_{,y} + \delta v_{,x}) \sigma_{xy}) dV\tag{2.8}$$

$$W_{\text{ext}} = \int_V \langle \delta u \rangle \{f_v\} dV + \int_{S_f} \langle \delta u \rangle \{f_s\} dS\tag{2.9}$$

L'équation 2.8 peut aussi être écrite sous la forme suivante :

$$W_{\text{int}} = \int_V \langle \delta \epsilon \rangle [H] \{ \epsilon \} dV\tag{2.10}$$

avec

$$\langle \delta \epsilon \rangle = \langle \delta \epsilon_{xx} \quad \delta \epsilon_{yy} \quad \delta \epsilon_{xy} \rangle\tag{2.11}$$

L'équation 2.10 est maintenant dans une forme appropriée et elle sera discrétisée par la méthode des éléments finis à la section suivante. Cette étape mènera à la formulation d'un élément de membrane triangulaire à trois nœuds (T3) ayant deux degrés de liberté par nœud, soit les déplacements u et v qui représentent les déplacements selon x et y respectivement.

2.1.4 Formulation de l'élément CST

2.1.4.1 Fonctions d'interpolation

Les fonctions d'interpolation pour l'élément CST (voir figure 2.1) sont définies de la façon suivante :

$$\begin{Bmatrix} u \\ v \end{Bmatrix} = [N]\{u_n\} = \begin{bmatrix} N_1 & 0 & N_2 & 0 & N_3 & 0 \\ 0 & N_1 & 0 & N_2 & 0 & N_3 \end{bmatrix} \{u_n\} \quad (2.12)$$

avec

$$\{u_n\} = \langle u_1 \ v_1 \ u_2 \ v_2 \ u_3 \ v_3 \rangle \quad (2.13)$$

$$\langle N \rangle = \langle 1 - \xi - \eta \ \xi \ \eta \rangle \quad (2.14)$$

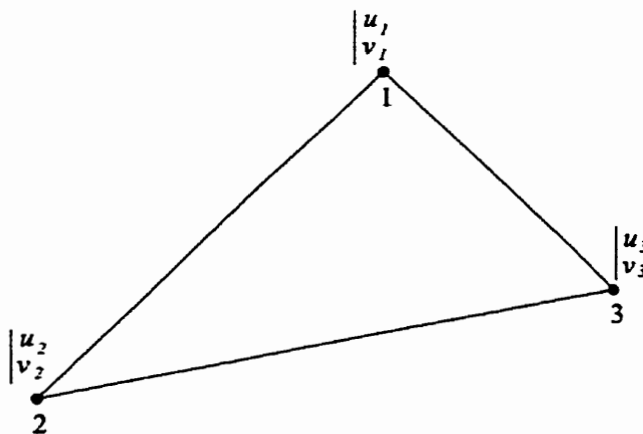


FIGURE 2.1: Élément de membrane

2.1.4.2 Évaluation de la matrice de rigidité

À l'aide de l'équation 2.3 et de l'équation 2.12, on peut écrire l'expression du vecteur des déformations sous la forme suivante :

$$\{\epsilon\} = \begin{Bmatrix} u_x \\ v_y \\ u_y + v_x \end{Bmatrix} = [B]\{u_n\} \quad (2.15)$$

avec

$$[B] = \begin{bmatrix} N_{1,x} & 0 & N_{2,x} & 0 & N_{3,x} & 0 \\ 0 & N_{1,y} & 0 & N_{2,y} & 0 & N_{3,y} \\ N_{1,y} & N_{1,x} & N_{2,y} & N_{2,x} & N_{3,y} & N_{3,x} \end{bmatrix} \quad (2.16)$$

Avec l'aide de la matrice jacobienne, il nous est possible d'exprimer les dérivées en x par des dérivées en ξ de la façon suivante :

$$\{\partial x\} = [j]\{\partial \xi\} = [J]^{-1}\{\partial \xi\} \quad (2.17)$$

On obtient alors :

$$\begin{Bmatrix} N_{i,x} \\ N_{i,y} \end{Bmatrix} = \frac{1}{J} \begin{bmatrix} J_{22} & -J_{12} \\ -J_{21} & J_{11} \end{bmatrix} \begin{Bmatrix} N_{i,\xi} \\ N_{i,\eta} \end{Bmatrix} \quad (2.18)$$

avec

$$J = J_{11}J_{22} - J_{12}J_{21} \quad (2.19)$$

$$\Rightarrow N_{i,x} = \frac{1}{J}(J_{22}N_{i,\xi} - J_{12}N_{i,\eta}) \quad N_{i,y} = \frac{1}{J}(-J_{21}N_{i,\xi} + J_{11}N_{i,\eta}) \quad (2.20)$$

Dans les équations précédentes, J représente le déterminant de la matrice jacobienne alors que les J_{ij} sont les termes de la matrice jacobienne. À l'aide des équations 2.9 et 2.10 et en tenant compte des résultats précédents, on peut définir l'expression de la forme intégrale faible sur chaque élément de la façon suivante :

$$W^e = \langle \delta u_n \rangle ([k]\{u_n\} - \{f_n\}) \quad (2.21)$$

avec

$$[k^e] = \int_{V^e} [B]^T [H] [B] dV \quad (2.22)$$

Puisque la matrice $[B]$ ne contient que des termes constants pour un élément, l'équation 2.22 peut s'écrire sous la forme suivante :

$$[k^e] = [B]^T \left(\int_{A^e} \underbrace{\int_{-\frac{h}{2}}^{+\frac{h}{2}} [H] dz}_{[H_m]} dA \right) [B] \quad (2.23)$$

$$\Rightarrow [k^e] = [B]^T \left(\int_{A^e} [H_m] dA \right) [B] \quad (2.24)$$

À l'aide de l'équation 2.24, il est facile de calculer la matrice de rigidité en membrane pour un élément e donné. La matrice $[H_m]$ nous est donnée par l'équation A.42 à la page 104. C'est à l'aide de cette matrice que la loi de comportement du matériau employé est prise en compte. Il est très important de remarquer que cet élément utilise une interpolation **linéaire** des déplacements. En effet, nous verrons au chapitre suivant que l'estimateur d'erreur utilisé suppose que la solution varie linéairement. La prochaine section présentera la description de l'élément utilisé pour les plaque fléchies.

2.2 Description de l'élément de flexion DST [7]

2.2.1 Relations déformations-déplacements

L'élément triangulaire à trois nœuds DST (*Discrete Shear Triangle*) généralise, en prenant compte des effets de cisaillement transverse, l'élément de plaque mince bien connu sous le nom de DKT (*Discrete Kirchhoff Triangle*). Ces deux éléments ont trois degrés de liberté par nœud : soit le déplacement latéral w et les rotations β_x et β_y . L'élément DST est utilisé pour modéliser les plaques minces ou épaisses

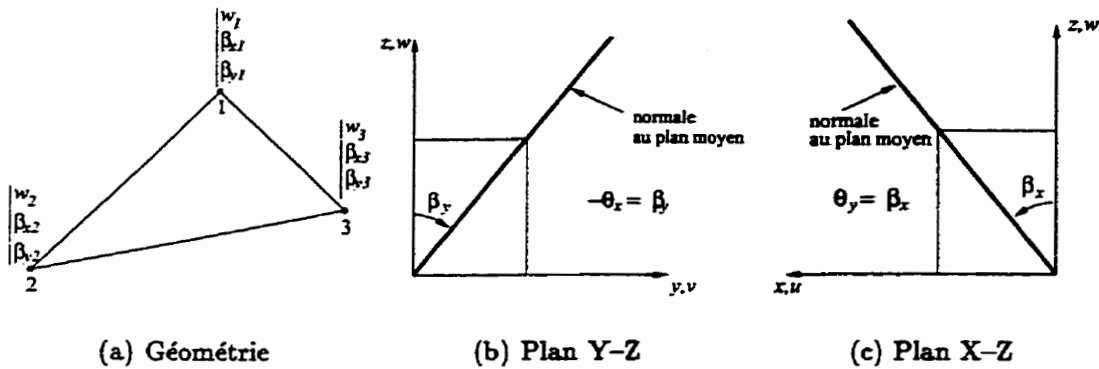


FIGURE 2.2: Élément triangulaire à 9 degrés de liberté

constituées d'un matériau isotrope, orthotrope ou laminé. Il sera superposé à l'élément décrit au paragraphe précédent pour permettre l'étude de coques à l'aide de facettes planes.

L'hypothèse de Mindlin/Reissner ou des plaques avec cisaillement transverse nous conduit à la définition des déplacements dans un système cartésien $x - y - z$. L'hypothèse de Mindlin-Reissner s'écrit de la façon suivante [4] :

“Les points matériels situés sur une normale à la surface moyenne non déformée restent sur une droite mais non nécessairement normale à la surface moyenne dans la configuration déformée.”

À partir de cette hypothèse, on peut définir les relations qui existent entre les déplacements u_q , v_q et w_q d'un point matériel q quelconque situé à l'intérieur d'une plaque soumise à un chargement de membrane et de flexion. On obtient donc à partir de cette hypothèse la cinématique suivante :

$$\begin{aligned}
 u_q(x, y, z) &= u(x, y) + z\beta_x(x, y) \\
 v_q(x, y, z) &= v(x, y) + z\beta_y(x, y) \\
 w_q(w, y, z) &= w(x, y)
 \end{aligned}
 \tag{2.25}$$

$$-\frac{h}{2} \leq z \leq \frac{h}{2}
 \tag{2.26}$$

Dans l'équation 2.25, x et y sont les coordonnées du plan moyen ($z = 0$) et z est la coordonnée suivant l'épaisseur h . Également, u , v et w sont les déplacements du plan moyen.

Dans le cadre des petites déformations et des petits déplacements entre les configurations initiale et finale, l'hypothèse de Mindlin/Reissner permet de définir les différentes composantes du vecteur des déformations :

$$\{\epsilon\} = \begin{Bmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ \gamma_{xy} \end{Bmatrix} = \begin{Bmatrix} u_{,x} \\ v_{,y} \\ u_{,y} + v_{,x} \end{Bmatrix} \quad (2.27)$$

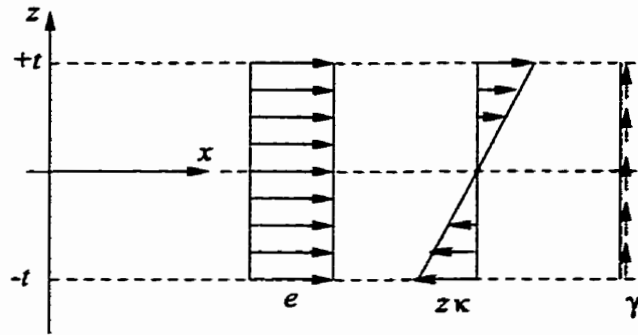
$$\{\gamma\} = \begin{Bmatrix} \gamma_{xz} \\ \gamma_{yz} \end{Bmatrix} = \begin{Bmatrix} u_{,z} + w_{,x} \\ v_{,z} + w_{,y} \end{Bmatrix} \quad (2.28)$$

Dans les équations 2.27 et 2.28, u , v et w représentent les fonctions cinématiques données par les relations de l'équation 2.25. Ainsi, à partir des relations de l'équation 2.25 et des équations 2.27 et 2.28 on obtient :

$$\{\epsilon\} = \{e\} + z\{\kappa\} = \begin{Bmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ \gamma_{xy} \end{Bmatrix} = \begin{Bmatrix} u_{,x} \\ v_{,y} \\ u_{,y} + v_{,x} \end{Bmatrix} + z \begin{Bmatrix} \beta_{x,z} \\ \beta_{y,y} \\ \beta_{x,y} + \beta_{y,x} \end{Bmatrix} \quad (2.29)$$

$$\{\gamma\} = \begin{Bmatrix} \gamma_{xz} \\ \gamma_{yz} \end{Bmatrix} = \begin{Bmatrix} \beta_x + w_{,x} \\ \beta_y + w_{,y} \end{Bmatrix} \quad (2.30)$$

Les vecteurs $\{e\}$ et $\{\kappa\}$ sont respectivement le vecteur des déformations de membrane pour le plan moyen et le vecteur des déformations de flexion (courbures). Les vecteurs $\{\epsilon\}$ et $\{\gamma\}$ sont les vecteurs des déformations de membrane pour un point situé dans l'épaisseur et des déformations de cisaillement transverse respectivement. Il est important de remarquer que les déformations $\{\epsilon\}$ sont linéaires en z et que les déformations de cisaillement transverse sont constantes à travers l'épaisseur comme le montre la figure 2.3.

FIGURE 2.3: Déformations e , $z\kappa$ et γ

2.2.2 Relation contraintes-déformations et efforts résultants

Pour une plaque constituée d'un matériau isotrope, orthotrope ou composite, nous obtenons les relations contraintes-déformations suivantes :

$$\{\sigma(z)\} = [H(z)] \{\epsilon(z)\} \quad \{\tau\} = [H_\tau(z)] \{\gamma\} \quad (2.31)$$

Les vecteurs $\{\epsilon\}$ et $\{\gamma\}$ nous sont donnés par les équations 2.29 et 2.30. Aussi, les efforts normaux $\{N\}$, les moments fléchissant $\{M\}$ et les efforts tranchants $\{T\}$ sont liés aux déformations $\{e\}$, $\{\kappa\}$ et $\{\gamma\}$ par les relations suivantes :

$$\{N\} = \begin{Bmatrix} N_x \\ N_y \\ N_{xy} \end{Bmatrix} = \int_{-t}^{+t} \begin{Bmatrix} \sigma_x \\ \sigma_y \\ \sigma_{xy} \end{Bmatrix} dz = [H_m] \{e\} + [H_{mf}] \{\kappa\} \quad (2.32)$$

$$\{M\} = \begin{Bmatrix} M_x \\ M_y \\ M_{xy} \end{Bmatrix} = \int_{-t}^{+t} \begin{Bmatrix} \sigma_x \\ \sigma_y \\ \sigma_{xy} \end{Bmatrix} z dz = [H_{mf}] \{e\} + [H_f] \{\kappa\} \quad (2.33)$$

$$\{T\} = \begin{Bmatrix} T_x \\ T_y \end{Bmatrix} = \int_{-t}^{+t} \begin{Bmatrix} \sigma_{xz} \\ \sigma_{yz} \end{Bmatrix} dz = [H_c] \{\gamma\} \quad (2.34)$$

Il est important de noter que la matrice $[H_c]$ tient compte des composantes de $[H_\tau]$ et des facteurs de correction de cisaillement transverse k (voir section A.3.1).

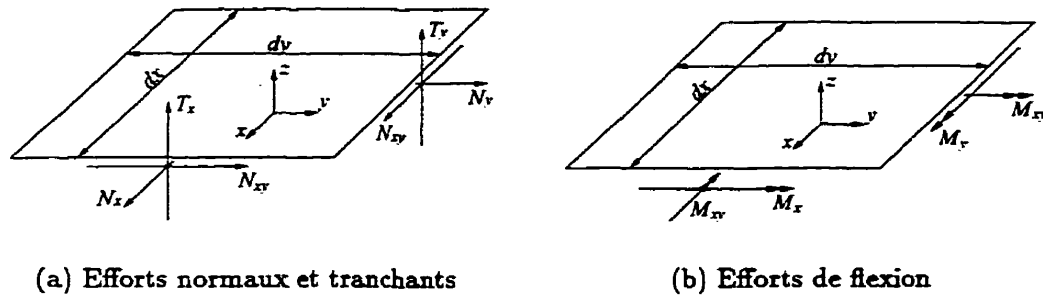


FIGURE 2.4: Efforts résultants sur un élément de plaque

2.2.3 Formulation détaillée de l'élément DST

Dans le cas où il n'y a pas de couplage membrane-flexion ($[H_{mf}] = 0$), les relations d'équilibre sur la surface moyenne peuvent être exprimées de la façon suivante :

$$\begin{aligned}
 T_{x,x} + T_{y,y} + p(x, y) &= 0 \\
 M_{x,x} + M_{zy,y} - T_x &= 0 \\
 M_{zy,x} + M_{y,y} - T_y &= 0
 \end{aligned}
 \tag{2.35}$$

où $p(x, y)$ est une pression agissant selon l'axe z .

En admettant l'existence d'une fonction scalaire dite énergie interne de déformation U^e pour un élément e donné telle que [7] :

$$U_T^e = U_f^e + U_c^e = \frac{1}{2} \langle u_n \rangle [k^e] \{u_n\}
 \tag{2.36}$$

avec

$$U_f^e = \frac{1}{2} \int_A \langle \kappa \rangle [H_f] \{ \kappa \} dA
 \tag{2.37}$$

$$U_c^e = \frac{1}{2} \int_A \langle \tilde{\gamma} \rangle [H_c] \{ \tilde{\gamma} \} dA
 \tag{2.38}$$

À l'aide des équations 2.36, 2.37 et 2.38, il nous est possible de calculer la matrice de rigidité $[k^e]$ pour un élément. Ici, U_f^e représente l'énergie interne de déformations de flexion et U_c^e l'énergie interne de déformation de cisaillement transverse. Le vecteur $\{ \tilde{\gamma} \}$ représente les déformations de cisaillement transverse indépendantes

ou de substitution. Les relations habituelles $\tilde{\gamma}_{xz} = \gamma_{zx}$ et $\tilde{\gamma}_{yz} = \gamma_{yz}$ seront exprimées sous forme discrète pour obtenir U_c^e en fonction des variables nodales $\{u_n\}$.

À l'aide de l'équation 2.34, nous pouvons redéfinir l'énergie interne de déformation de cisaillement transverse (équation 2.38) de la façon suivante :

$$U_c^e = \frac{1}{2} \int_A \langle T \rangle [H_c]^{-1} \{T\} dA \quad (2.39)$$

Nous savons aussi à l'aide de l'équation 2.33 et des relations d'équilibre 2.35 que :

$$\{T\} = \begin{Bmatrix} T_x \\ T_y \end{Bmatrix} = \begin{Bmatrix} M_{x,x} + M_{xy,y} \\ M_{xy,x} + M_{y,y} \end{Bmatrix} \quad (2.40)$$

avec

$$\{M\} = [H_f] \{\kappa\} \quad (2.41)$$

Ainsi, les déformations $\tilde{\gamma}_{xz}$ et $\tilde{\gamma}_{yz}$ peuvent être définies en fonction des dérivées secondes de β_x et β_y par l'intermédiaire de deux équations d'équilibre interne et d'une loi de comportement en flexion. En multipliant les différents termes de l'équation 2.33 et toujours en considérant qu'il n'y a pas de couplage membrane-flexion nous obtenons :

$$\{M\} = \begin{Bmatrix} M_x \\ M_y \\ M_{xy} \end{Bmatrix} = \begin{Bmatrix} H_{11}\beta_{x,x} + H_{12}\beta_{y,y} + H_{13}(\beta_{x,y} + \beta_{y,x}) \\ H_{21}\beta_{x,x} + H_{22}\beta_{y,y} + H_{23}(\beta_{x,y} + \beta_{y,x}) \\ H_{31}\beta_{x,x} + H_{32}\beta_{y,y} + H_{33}(\beta_{x,y} + \beta_{y,x}) \end{Bmatrix} \quad (2.42)$$

Dans l'équation 2.42, les H_{ij} représentent les termes de la matrices $[H_f]$. Avec l'aide des équations 2.42, nous pouvons maintenant évaluer explicitement l'équation 2.40 et ainsi obtenir l'expression de l'effort résultant $\{T\}$ en fonction des différents termes de la matrice $[H_f]$ et des dérivées secondes des rotations β_x et β_y .

$$\{T\} = \begin{Bmatrix} T_x \\ T_y \end{Bmatrix} = \begin{Bmatrix} H_{11}\beta_{x,xx} + H_{12}\beta_{y,yx} + H_{13}(\beta_{x,yx} + \beta_{y,xx}) + \\ H_{31}\beta_{x,xy} + H_{32}\beta_{y,yy} + H_{33}(\beta_{x,yy} + \beta_{y,xy}) \\ H_{31}\beta_{x,xx} + H_{32}\beta_{y,yx} + H_{33}(\beta_{x,yx} + \beta_{y,xx}) + \\ H_{21}\beta_{x,xy} + H_{22}\beta_{y,yy} + H_{23}(\beta_{x,yy} + \beta_{y,xy}) \end{Bmatrix} \quad (2.43)$$

ou sous forme matricielle :

$$\{T\} = [\bar{H}_f] \{\beta_{,xx}\} \quad (2.44)$$

$$\{T\} = \begin{bmatrix} H_{11} & H_{33} & H_{13} + H_{31} & H_{13} & H_{32} & H_{12} + H_{33} \\ H_{31} & H_{23} & H_{33} + H_{21} & H_{33} & H_{22} & H_{32} + H_{23} \end{bmatrix} \{\beta_{,xx}\} \quad (2.45)$$

$$\text{avec } \langle \beta_{,xx} \rangle = \langle \beta_{x,xx} \quad \beta_{x,yy} \quad \beta_{x,xy} \quad \beta_{y,xx} \quad \beta_{y,yy} \quad \beta_{y,xy} \rangle \quad (2.46)$$

Aussi, les règles de dérivation en chaîne nous permettent de calculer les dérivées secondes en x d'une fonction à partir de ses dérivées en ξ du système de référence de l'élément.

$$\begin{Bmatrix} \frac{\partial^2}{\partial x^2} \\ \frac{\partial^2}{\partial y^2} \\ \frac{\partial^2}{\partial x \partial y} \end{Bmatrix} = [T_1] \begin{Bmatrix} \frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial \eta} \end{Bmatrix} + [T_2] \begin{Bmatrix} \frac{\partial^2}{\partial \xi^2} \\ \frac{\partial^2}{\partial \eta^2} \\ \frac{\partial^2}{\partial \xi \partial \eta} \end{Bmatrix} \quad (2.47)$$

Puisque les termes de la matrice jacobienne $[J]$ sont constants pour un triangle à trois nœuds, la matrice $[T_1]$ est nulle et donc les dérivées secondes en x peuvent être exprimées à partir de la matrice $[T_2]$ seulement.

$$\{\beta_{,xx}\} = [T_2] \{\beta_{,\xi\xi}\} \quad (2.48)$$

$$\text{avec } \langle \beta_{,\xi\xi} \rangle = \langle \beta_{x,\xi\xi} \quad \beta_{x,\eta\eta} \quad \beta_{x,\xi\eta} \quad \beta_{y,\xi\xi} \quad \beta_{y,\eta\eta} \quad \beta_{y,\xi\eta} \rangle \quad (2.49)$$

$$[T_2] = \begin{bmatrix} j_{11}^2 & j_{12}^2 & 2j_{11}j_{12} \\ j_{21}^2 & j_{22}^2 & 2j_{21}j_{22} \\ 2j_{11}j_{21} & 2j_{12}j_{22} & j_{11}j_{22} + j_{12}j_{21} \end{bmatrix} \quad (2.50)$$

Ici, les j_{ij} représentent les termes de la matrice jacobienne inversée.

Avant de pouvoir effectuer la dérivation des rotations β_x et β_y , il faut déterminer une approximation adéquate pour ces fonctions. L'approximation retenue est du type C^0 (quadratique incomplète) sous la forme :

$$\begin{aligned}\beta_x &= \langle N \rangle \{ \beta_{xn} \} + \langle P_x \rangle \{ \alpha_k \} \\ \beta_y &= \langle N \rangle \{ \beta_{yn} \} + \langle P_y \rangle \{ \alpha_k \}\end{aligned}\tag{2.51}$$

avec

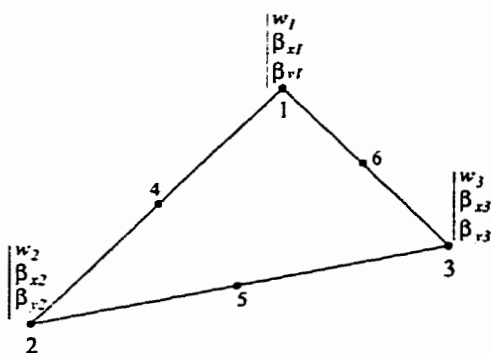
$$\langle N \rangle = \langle 1 - \xi - \eta \quad \xi \quad \eta \rangle\tag{2.52}$$

$$\langle \alpha_k \rangle = \langle \alpha_4 \quad \alpha_5 \quad \alpha_6 \rangle\tag{2.53}$$

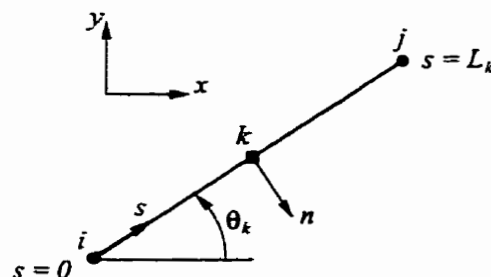
$$\langle P_x \rangle = \langle P_4 C_4 \quad P_5 C_5 \quad P_6 C_6 \rangle\tag{2.54}$$

$$\langle P_y \rangle = \langle P_4 S_4 \quad P_5 S_5 \quad P_6 S_6 \rangle\tag{2.55}$$

$$\langle P_k \rangle = \langle 4\xi(1 - \xi - \eta) \quad 4\xi\eta \quad 4\eta(1 - \xi - \eta) \rangle\tag{2.56}$$



(a) Élément réel



(b) Système de coordonnées local sur un côté

FIGURE 2.5: Élément réel et système de coordonnées local sur un côté

Les coefficients C_k et S_k sont les cosinus directeurs du côté k associés aux nœuds i et j . À partir des coordonnées des nœuds, on peut définir ces coefficients de la façon suivante (voir figure 2.5(b)) :

$$\begin{aligned} C_k &= \cos \theta_k = \frac{x_{ji}}{L_k} = \frac{x_j - x_i}{L_k} \\ S_k &= \sin \theta_k = \frac{y_{ji}}{L_k} = \frac{y_j - y_i}{L_k} \end{aligned} \quad (2.57)$$

avec

$$L_k^2 = x_{ji}^2 + y_{ji}^2 \quad (2.58)$$

Les coefficients α_k sont les variables associées à la représentation quadratique de β_x et β_y . Par exemple, sur le côté 1-2 associé à $k = 4$, on obtient la relation suivante :

$$\begin{aligned} (\beta_x)_{1-2} &= (1 - \xi - \eta)\beta_{x1} + \xi\beta_{x2} + 4\xi(1 - \xi - \eta)C_4\alpha_4 \\ &= (1 - \xi)\beta_{x1} + \xi\beta_{x2} + 4\xi(1 - \xi)C_4\alpha_4 \end{aligned}$$

De la même manière, on obtient β_y sur le côté 1-2 :

$$(\beta_y)_{1-2} = (1 - \xi)\beta_{y1} + \xi\beta_{y2} + 4\xi(1 - \xi)S_4\alpha_4$$

Il est possible de *tourner* les rotations β_x et β_y pour obtenir les rotations selon le système s et n défini à la figure 2.5 pour ainsi obtenir les rotations β_s et β_n . La relation entre les rotations β_x et β_y et les rotations β_s et β_n est :

$$\begin{Bmatrix} \beta_s \\ \beta_n \end{Bmatrix} = \begin{bmatrix} \cos \theta_k & \sin \theta_k \\ \sin \theta_k & -\cos \theta_k \end{bmatrix} \begin{Bmatrix} \beta_x \\ \beta_y \end{Bmatrix} = \begin{bmatrix} C_k & S_k \\ S_k & -C_k \end{bmatrix} \begin{Bmatrix} \beta_x \\ \beta_y \end{Bmatrix} \quad (2.59)$$

Maintenant, la rotation β_s sur le côté 1-2 devient :

$$\begin{aligned} (\beta_s)_{1-2} &= C_4\beta_x + S_4\beta_y \\ (\beta_s)_{1-2} &= (1 - \xi)\beta_{x1}C_4 + \xi\beta_{x2}C_4 + 4\xi(1 - \xi)C_4^2\alpha_4 + \\ &\quad (1 - \xi)\beta_{y1}S_4 + \xi\beta_{y2}S_4 + 4\xi(1 - \xi)S_4^2\alpha_4 \\ (\beta_s)_{1-2} &= (1 - \xi)(\beta_{x1}C_4 + \beta_{y1}S_4) + \xi(\beta_{x2}C_4 + \beta_{y2}S_4) + \\ &\quad 4\xi(1 - \xi)\alpha_4(C_4^2 + S_4^2) \end{aligned}$$

$$(\beta_s)_{1-2} = (1 - \xi)\beta_{s1} + \xi\beta_{s2} + 4\xi(1 - \xi)\alpha_4 \quad (\text{quadratique en } \xi)$$

Aussi, la rotation β_n sur le côté 1-2 devient :

$$(\beta_n)_{1-2} = S_4\beta_x - C_4\beta_y$$

$$\begin{aligned} (\beta_n)_{1-2} &= (1 - \xi)\beta_{x1}S_4 + \xi\beta_{x2}S_4 + 4\xi(1 - \xi)S_4C_4\alpha_4 - \\ &\quad (1 - \xi)\beta_{y1}C_4 - \xi\beta_{y2}C_4 - 4\xi(1 - \xi)S_4C_4\alpha_4 \end{aligned}$$

$$(\beta_n)_{1-2} = (1 - \xi)(\beta_{x1}S_4 - \beta_{y1}C_4) + \xi(\beta_{x2}S_4 - \beta_{y2}C_4)$$

$$(\beta_n)_{1-2} = (1 - \xi)\beta_{n1} + \xi\beta_{n2} \quad (\text{linéaire en } \xi)$$

De façon plus générale, nous pouvons noter que sur un côté dont les nœuds d'extrémités sont i et j et dont le nœud milieu est k , les rotations β_s et β_n sont :

$$\beta_s = (1 - s')\beta_{si} + s'\beta_{sj} + 4s'(1 - s')\alpha_k \quad (2.60)$$

$$\beta_n = (1 - s')\beta_{ni} + s'\beta_{nj} \quad (2.61)$$

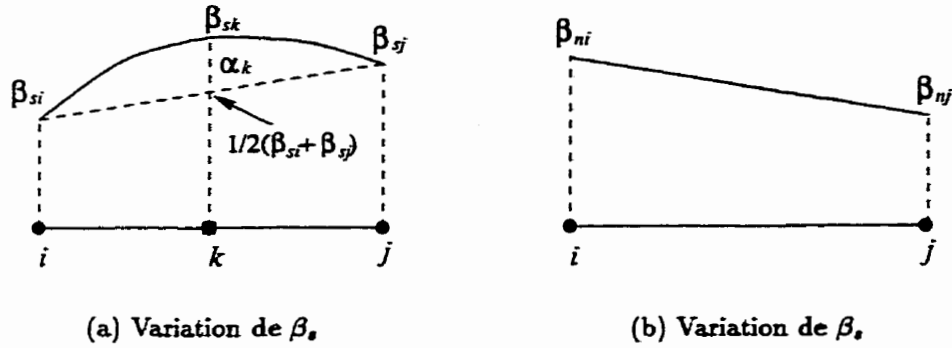
avec

$$0 \leq s' = \frac{s}{L_k} \leq 1$$

On voit très bien que les approximations retenues pour β_x et β_y sont telles que β_s est quadratique et β_n linéaire sur les côtés. Cette constatation peut très bien être visualisée sur la figure 2.6 On remarque à partir des équations 2.60 et 2.61 qu'au nœud milieu k ($s' = 1/2$), l'expression de β_s et β_n deviennent :

$$\begin{aligned} \beta_{sk} &= (1 - s')\beta_{si} + s'\beta_{sj} + 4s'(1 - s')\alpha_k \\ &= \frac{1}{2}(\beta_{si} + \beta_{sj}) + \alpha_k \end{aligned}$$

$$\begin{aligned} \beta_n &= (1 - s')\beta_{ni} + s'\beta_{nj} \\ &= \frac{1}{2}(\beta_{ni} + \beta_{nj}) \end{aligned}$$


 FIGURE 2.6: Variation des rotations β_s et β_n sur les côtés

Il nous est maintenant possible de poursuivre l'évaluation de l'équation 2.48 et ainsi définir les dérivées secondes de β par rapport au système $\xi\eta$ de l'élément de référence puisque l'on connaît maintenant les fonctions d'approximation des rotations β . Avec l'aide de l'équation 2.51, on a :

$$\begin{aligned} \beta_{x,\xi\xi} &= \langle N_{,\xi\xi} \rangle \{ \beta_{zn} \} + \langle P_x, \xi\xi \rangle \{ \alpha_k \} \\ &= -8C_4 \alpha_4 \end{aligned} \quad (2.62)$$

En dérivant ainsi β_x et β_y par rapport aux autres directions du système $\xi\eta$, on obtient l'expression suivante sous forme matricielle :

$$\{ \beta_{,\xi\xi} \} = [T_\alpha] \{ \alpha_k \} = \begin{bmatrix} -8C_4 & 0 & 0 \\ 0 & 0 & -8C_6 \\ -4C_4 & 4C_5 & -4C_6 \\ -8S_4 & 0 & 0 \\ 0 & 0 & -8S_6 \\ -4S_4 & 4S_5 & -4S_6 \end{bmatrix} \begin{Bmatrix} \alpha_4 \\ \alpha_5 \\ \alpha_6 \end{Bmatrix} \quad (2.63)$$

Avec l'aide des équations 2.48 et 2.63, l'équation 2.45 prend maintenant la forme suivante :

$$\{ T \} = [B_{\alpha\alpha}] \{ \alpha_k \} = [\bar{H}_f] [T_2] [T_\alpha] \{ \alpha_k \} \quad (2.64)$$

Il nous est maintenant possible d'exprimer l'énergie interne de déformation de cisaillement transverse en fonction des variables nodales α_k sur les côtés. L'équation 2.39 devient ainsi :

$$U_c^e = \frac{1}{2} \int_A \langle \alpha_k \rangle [B_{c\alpha}]^T [H_c]^{-1} [B_{c\alpha}] \{ \alpha_k \} dA \quad (2.65)$$

Pour un élément donné, les matrices $[B_{c\alpha}]$ et $[H_c]$ sont constantes. Nous pouvons donc écrire l'équation 2.65 sous la forme suivante :

$$U_c^e = \frac{1}{2} \langle \alpha_k \rangle [k_{c\alpha}]^T \{ \alpha_k \} \quad (2.66)$$

avec

$$[k_{c\alpha}] = A [B_{c\alpha}]^T [H_c]^{-1} [B_{c\alpha}] \quad (2.67)$$

Dans la dernière équation, A représente l'aire de l'élément réel. La matrice $[B_{c\alpha}]$ est donnée avec l'aide de l'équation 2.64 et la matrice $[H_c]$ à l'aide de l'équation A.67 de la page 107. À partir de toutes les relations données jusqu'à maintenant, il est possible de prendre en compte le cisaillement transverse sur les différents côtés d'un élément. Afin de générer un nombre suffisant d'équations pour éliminer les variables non nodales α_k , on définit la relation suivante :

$$\int_0^{L_k} (\gamma_s - \tilde{\gamma}_s) ds = 0 \quad (2.68)$$

Cette équation nous indique que la différence entre les déformations réelles γ_s et celles exprimées sous forme discrète $\tilde{\gamma}_s$ doivent s'annuler sur chacun des côtés d'un élément. L'expression de γ_s sur un côté en fonction des déformations γ_{xz} et γ_{yz} nous est donné par :

$$\gamma_s = C_k \gamma_{xz} + S_k \gamma_{yz} = w_{,s} + \beta_s \quad (2.69)$$

On sait par contre à partir de l'équation 2.34 que :

$$\{ \tilde{\gamma} \} = \left\{ \begin{array}{c} \tilde{\gamma}_{xz} \\ \tilde{\gamma}_{yz} \end{array} \right\} = [H_c]^{-1} \{ T(\beta_x, \beta_y) \} \quad (2.70)$$

On obtient donc avec l'aide de l'équation 2.64 :

$$\{\tilde{\gamma}_s\} = \langle C_k \ S_k \rangle \{\tilde{\gamma}\} = \langle C_k \ S_k \rangle [H_c]^{-1} [B_{\alpha\alpha}] \{\alpha_k\} \quad (2.71)$$

L'équation 2.68 peut maintenant être évaluée complètement. On obtient alors :

$$\int_0^{L_k} (\gamma_s - \tilde{\gamma}_s) ds = \int_0^{L_k} ((w_{,s} + \beta_s) - \tilde{\gamma}_s) ds \quad (2.72)$$

$$\int_0^{L_k} \beta_s ds + \int_0^{L_k} w_{,s} ds = \int_0^{L_k} \tilde{\gamma}_s (\beta_x, \beta_y) ds = 0 \quad (2.73)$$

L'équation 2.60 nous permet d'obtenir :

$$w \Big|_0^{L_k} + \int_0^{L_k} ((1-s')\beta_{si} + s'\beta_{sj} + 4s'(1-s')\alpha_k) ds = \int_0^{L_k} \underbrace{\tilde{\gamma}_s}_{\text{const.}} ds \quad (2.74)$$

En prenant le changement de variable suivant :

$$s = s' L_k \quad \Rightarrow \quad ds = L_k ds' \quad (2.75)$$

on obtient :

$$w_j - w_i + L_k \int_0^1 (1-s')\beta_{si} + s'\beta_{sj} + 4s'(1-s')\alpha_k ds = L_k \tilde{\gamma}_{sk} \quad (2.76)$$

$$w_j - w_i + L_k \left(\left(s' - \frac{s'^2}{2} \right) \beta_{si} + \frac{s'^2}{2} \beta_{sj} + \left(\frac{4s'^2}{2} - \frac{4s'^3}{3} \right) \alpha_k \right) \Big|_0^1 = L_k \tilde{\gamma}_{sk} \quad (2.77)$$

$$\Rightarrow \quad w_j - w_i + L_k \left(\frac{1}{2} \beta_{si} + \frac{1}{2} \beta_{sj} + \frac{2}{3} \alpha_k \right) = L_k \tilde{\gamma}_{sk} \quad (2.78)$$

À l'aide de l'équation 2.59, il nous est possible d'exprimer β_{si} et β_{sj} de l'équation 2.78 en fonction des β_x et β_y . On obtient alors :

$$w_j - w_i + \frac{L_k}{2} (C_k \beta_{xi} + S_k \beta_{yi} + C_k \beta_{xj} + S_k \beta_{yj}) + \frac{2}{3} \alpha_k L_k = L_k \tilde{\gamma}_{sk} \quad (2.79)$$

L'équation 2.79 précédente peut aussi être réécrite sous la forme matricielle suivante :

$$[A_w]\{u_n\} = \frac{2}{3}\alpha_k L_k - L_k \bar{\gamma}_{sk} \quad (2.80)$$

avec

$$[A_w] = -\frac{1}{2} \left[\begin{array}{ccc|ccc} -2 & L_4 C_4 & L_4 S_4 & 2 & L_4 C_4 & L_4 S_4 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2 & L_5 C_5 & L_5 S_5 & 2 & L_5 C_5 & L_5 S_5 \\ 2 & L_6 C_6 & L_6 S_6 & 0 & 0 & 0 & -2 & L_6 C_6 & L_6 S_6 \end{array} \right] \quad (2.81)$$

et

$$\langle u_n \rangle = \langle \dots w_i \beta_{xi} \beta_{yi} \dots \rangle \quad (2.82)$$

À l'aide de l'équation 2.71, nous pouvons maintenant évaluer le membre de droite des équations 2.79 ou 2.80 :

$$\frac{2}{3}\alpha_k L_k - L_k \bar{\gamma}_{sk} = \frac{2}{3}\alpha_k L_k - \langle C_k \ S_k \rangle [H_c]^{-1} [B_{c\alpha}] \alpha_k \quad (2.83)$$

soit sous forme matricielle :

$$\frac{2}{3}\alpha_k L_k - L_k \bar{\gamma}_{sk} = [A_\alpha]\{\alpha_k\} \quad (2.84)$$

avec

$$[A_\alpha] = \frac{2}{3} \left[\begin{array}{ccc} L_4 & 0 & 0 \\ 0 & L_5 & 0 \\ 0 & 0 & L_6 \end{array} \right] - \left[\begin{array}{cc} L_4 C_4 & L_4 S_4 \\ L_5 C_5 & L_5 S_5 \\ L_6 C_6 & L_6 S_6 \end{array} \right] [H_c]^{-1} [B_{c\alpha}] \quad (2.85)$$

On obtient finalement

$$[A_\alpha]\{\alpha_k\} = [A_w]\{u_n\} \quad (2.86)$$

Cette dernière équation nous permet de définir les variables nodales sur les côtés $\{\alpha_k\}$ en fonction des variables nodales finales $\{u_n\}$:

$$\{\alpha_k\} = [A_\alpha]^{-1}[A_w]\{u_n\} = [A_n]\{u_n\} \quad (2.87)$$

Ainsi, les variables non nodales sur les côtés $\{\alpha_k\}$ sont supprimées des équations. Les approximations de β_x et β_y deviennent donc :

$$\beta_x = \langle N \rangle \{\beta_{xn}\} + \langle P_x \rangle [A_n] \{u_n\} = \langle N_x \rangle \{u_n\} \quad (2.88)$$

$$\beta_y = \langle N \rangle \{\beta_{yn}\} + \langle P_y \rangle [A_n] \{u_n\} = \langle N_y \rangle \{u_n\} \quad (2.89)$$

On remarque qu'aucune approximation des variables nodales w_i n'a été définie. En effet, elles apparaissent naturellement lors de l'évaluation de l'équation 2.68. La nature des variables nodales finales (trois valeurs de w aux sommets) nous permet de considérer w linéaire sur l'élément pour définir le vecteur des charges équivalentes et la matrice masse. Par contre, il est très important de noter que w est *compatible* avec une approximation **cubique** des déplacements étant donné l'utilisation d'une approximation quadratique des rotations β_x et β_y . Par contre, cette approximation est cubique de continuité C^1 , i.e. en utilisant des fonctions d'interpolation de type Hermite.

À l'aide de l'approximation de β_x et β_y , il nous est maintenant possible de calculer les termes de $\{\kappa\}$ pour pouvoir évaluer l'énergie interne de flexion à partir des variables nodales finales (trois valeurs à chaque sommet). Ainsi, le vecteur $\{\kappa\}$:

$$\{\kappa\} = \left\{ \begin{array}{c} \beta_{x,x} \\ \beta_{y,y} \\ \beta_{x,y} + \beta_{y,x} \end{array} \right\} \quad (2.90)$$

Pour permettre l'évaluation de l'équation 2.90, il est nécessaire de connaître $\beta_{x,x}$, $\beta_{x,y}$, $\beta_{y,y}$ et $\beta_{y,x}$. Les équations qui vont suivre donne un exemple de calcul pour $\beta_{x,x}$. À l'aide de l'équation 2.51, on a :

$$\beta_x = \langle N_1 \ N_2 \ N_3 \rangle \{\beta_{xn}\} + \langle P_x \rangle \{\alpha_k\} \quad (2.91)$$

$$\beta_x = N_1\beta_{x1} + N_2\beta_{x2} + N_3\beta_{x3} + P_{x4}\alpha_4 + P_{x5}\alpha_5 + P_{x6}\alpha_6 \quad (2.92)$$

$$\Rightarrow \beta_{x,x} = \frac{\partial N_1}{\partial x}\beta_{x1} + \frac{\partial N_2}{\partial x}\beta_{x2} + \frac{\partial N_3}{\partial x}\beta_{x3} + \frac{\partial P_{x4}}{\partial x}\alpha_4 + \frac{\partial P_{x5}}{\partial x}\alpha_5 + \frac{\partial P_{x6}}{\partial x}\alpha_6 \quad (2.93)$$

Les dérivées en x peuvent être obtenues des dérivées en ξ du système de référence à l'aide de la relation suivante :

$$\left\{ \begin{array}{c} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial x} \end{array} \right\} = \left[\begin{array}{cc} \frac{\partial \xi}{\partial x} & \frac{\partial \eta}{\partial x} \\ \frac{\partial \xi}{\partial y} & \frac{\partial \eta}{\partial y} \end{array} \right] \left\{ \begin{array}{c} \frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial \eta} \end{array} \right\} = \left[\begin{array}{cc} j_{11} & j_{12} \\ j_{21} & j_{22} \end{array} \right] \left\{ \begin{array}{c} \frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial \eta} \end{array} \right\} \quad (2.94)$$

Nous obtenons donc :

$$\begin{aligned} \beta_{x,x} = & \left(j_{11} \frac{\partial N_1}{\partial \xi} + j_{12} \frac{\partial N_1}{\partial \eta} \right) \beta_{x1} + \left(j_{11} \frac{\partial N_2}{\partial \xi} + j_{12} \frac{\partial N_2}{\partial \eta} \right) \beta_{x2} + \\ & \left(j_{11} \frac{\partial N_3}{\partial \xi} + j_{12} \frac{\partial N_3}{\partial \eta} \right) \beta_{x3} + \left(j_{11} \frac{\partial P_{x4}}{\partial \xi} + j_{12} \frac{\partial P_{x4}}{\partial \eta} \right) \alpha_4 + \\ & \left(j_{11} \frac{\partial P_{x5}}{\partial \xi} + j_{12} \frac{\partial P_{x5}}{\partial \eta} \right) \alpha_5 + \left(j_{11} \frac{\partial P_{x6}}{\partial \xi} + j_{12} \frac{\partial P_{x6}}{\partial \eta} \right) \alpha_6 \end{aligned} \quad (2.95)$$

Finalement, pour le calcul de $\beta_{x,x}$ l'équation suivante :

$$\begin{aligned} \beta_{x,x} = & (-j_{11} - j_{12})\beta_{x1} + (j_{11} + 0)\beta_{x2} + 0 + j_{12})\beta_{x3} + \\ & (j_{11}P_{x4,\xi} + j_{12}P_{x4,\eta})\alpha_4 + (j_{11}P_{x5,\xi} + j_{12}P_{x5,\eta})\alpha_5 + \\ & (j_{11}P_{x6,\xi} + j_{12}P_{x6,\eta})\alpha_6 \end{aligned} \quad (2.96)$$

De la même manière nous pouvons calculer $\beta_{y,y}$, $\beta_{x,y}$ et $\beta_{y,x}$. Nous obtenons finalement :

$$\{\kappa\} = [B_f] \left\{ \begin{array}{c} u_n \\ \alpha_k \end{array} \right\} = \left[\begin{array}{c|c} B_{f\beta} & B_{f\alpha} \end{array} \right] \left\{ \begin{array}{c} u_n \\ \alpha_k \end{array} \right\} \quad (2.97)$$

avec

$$[B_{f\beta}] = \begin{bmatrix} 0 & -(j_{11} + j_{12}) & 0 & 0 & j_{11} & 0 & 0 & j_{12} & 0 \\ 0 & 0 & -(j_{21} + j_{22}) & 0 & 0 & j_{21} & 0 & 0 & j_{22} \\ 0 & -(j_{21} + j_{22}) & -(j_{11} + j_{12}) & 0 & j_{21} & j_{11} & 0 & j_{22} & j_{12} \end{bmatrix} \quad (2.98)$$

$$[B_{f\alpha}] = \begin{bmatrix} \dots & j_{11}P_{zk,\xi} + j_{12}P_{zk,\eta} & \dots \\ \dots & j_{21}P_{yk,\xi} + j_{22}P_{yk,\eta} & \dots \\ \dots & (j_{21}P_{zk,\xi} + j_{22}P_{zk,\eta} + j_{11}P_{yk,\xi} + j_{12}P_{yk,\eta}) & \dots \end{bmatrix} \quad (2.99)$$

Il est important de noter ici que les inconnus nodales w_i n'ont aucun terme associé dans la matrice $[B_f]$. Ceux-ci sont introduits naturellement lors de l'évaluation de l'équation 2.68. Il nous est maintenant possible d'évaluer l'énergie interne de flexion en fonction des variables nodales β_x , β_y et α_k pour ainsi obtenir la matrice de rigidité associée aux déformations de flexion.

$$U_f^e = \frac{1}{2} \int_A \langle \kappa \rangle [H_f] \{ \kappa \} dA = \frac{1}{2} \langle u_n \quad \alpha_k \rangle [k_{f\alpha}] \begin{Bmatrix} u_n \\ \alpha_k \end{Bmatrix} \quad (2.100)$$

avec

$$[k_{f\alpha}]_{12 \times 12} = \begin{bmatrix} [k_{11}]_{9 \times 9} & [k_{12}]_{9 \times 3} \\ [k_{12}]_{3 \times 9}^T & [k_{22}]_{3 \times 3} \end{bmatrix} = \int_A [B_f]^T [H_f] [B_f] dA \quad (2.101)$$

La matrice de rigidité $[H_{f\alpha}]$ peut être intégrée exactement avec trois points de *Hammer* si la matrice $[H_f]$ est constante. Nous avons maintenant toutes les équations nécessaires pour évaluer l'énergie interne totale de déformation de flexion en fonction des variables nodales u_n et α_k . L'énergie interne totale de déformation de flexion est :

$$U_T^e = U_f^e + U_c^e \quad (2.102)$$

avec

$$U_f^e = \frac{1}{2} \langle u_n \quad \alpha_k \rangle \begin{bmatrix} [k_{11}]_{9 \times 9} & [k_{12}]_{9 \times 3} \\ [k_{12}]_{3 \times 9}^T & [k_{22}]_{3 \times 3} \end{bmatrix} \begin{Bmatrix} u_n \\ \alpha_k \end{Bmatrix} \quad (2.103)$$

$$U_c^e = \frac{1}{2} \langle \alpha_k \rangle [k_{c\alpha}] \{ \alpha_k \} \quad (2.104)$$

Avec l'aide de l'équation 2.87, il nous est possible de supprimer les variables non nodales situées sur chaque côté d'un élément à l'aide d'une *condensation statique*. On obtient alors :

$$U_f^e = \frac{1}{2} \langle u_n \quad u_n [A_n]^T \rangle \begin{bmatrix} [k_{11}]_{9 \times 9} & [k_{12}]_{9 \times 3} \\ [k_{12}]_{3 \times 9}^T & [k_{22}]_{3 \times 3} \end{bmatrix} \begin{Bmatrix} u_n \\ [A_n] u_n \end{Bmatrix} \quad (2.105)$$

$$U_c^e = \frac{1}{2} \langle u_n [A_n]^T \rangle [k_{c\alpha}] \{ [A_n] u_n \} \quad (2.106)$$

En isolant les variables nodales $\{u_n\}$, on obtient :

$$U_f^e = \frac{1}{2} \langle u_n \rangle \langle 1 \quad [A_n]^T \rangle \begin{bmatrix} [k_{11}]_{9 \times 9} & [k_{12}]_{9 \times 3} \\ [k_{12}]_{3 \times 9}^T & [k_{22}]_{3 \times 3} \end{bmatrix} \begin{Bmatrix} \{1\} \\ [A_n] \end{Bmatrix} \{u_n\} \quad (2.107)$$

$$U_c^e = \frac{1}{2} \langle u_n \rangle \langle [A_n]^T \rangle [k_{c\alpha}] \{ [A_n] \} \{u_n\} \quad (2.108)$$

À l'aide des équations 2.107 et 2.108, la matrice de rigidité $[k^e]$ pour un élément devient finalement :

$$[k^e] = \underbrace{[k_{11}] + [A_n]^T [k_{12}] [A_n] + [A_n]^T [k_{12}]^T + [A_n]^T [k_{22}] [A_n]}_{\text{termes de flexion}} + \underbrace{[A_n]^T [k_{c\alpha}] [A_n]}_{\text{terme de cisaillement}} \quad (2.109)$$

Si les effets de cisaillement transverse sont négligeables (dans le cas de plaque très mince), l'équation 2.28 tend vers zéro et l'élément DST converge naturellement vers l'élément DKT ce qui a pour effet d'éviter le blocage en cisaillement.

2.3 Assemblage

Dans cette section, nous présentons l'assemblage au niveau global de l'élément appelé DST18 où la partie membrane est représentée par l'élément linéaire CST (voir section 2.1) et la partie flexion par l'élément DST (voir section 2.2). Les variables nodales, après assemblage, sont les déplacements U, V, W des trois nœuds et les rotations $\theta_X, \theta_Y, \theta_Z$ autour des axes globaux X, Y, Z (voir figure 2.7). Le lecteur est invité à consulter [8, §6.4] pour plus d'information sur le sujet.

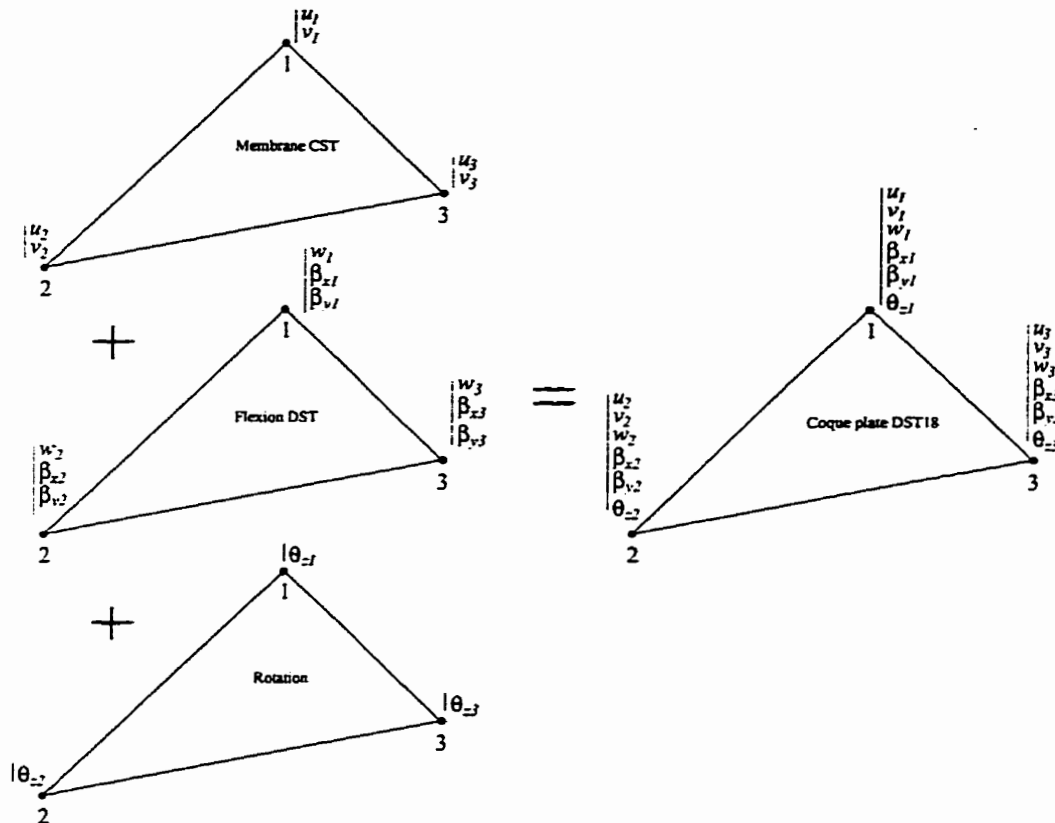


FIGURE 2.7: Construction de l'élément DST18

Il est important de noter que nous considérons le vecteur normal $\{n\}$ perpendiculaire à la surface moyenne des éléments. De cette façon, $\{n\}$ est discontinu d'un élément à l'autre.

2.3.1 Définition du repère local

Afin de pouvoir calculer la matrice de rigidité d'un élément orienté dans l'espace 3D, nous devons d'abord calculer la matrice $[k^e]$ dans le repère local de l'élément pour ensuite *tourner* cette matrice vers le système global.

La surface de référence de l'élément est définie par :

$$\{x_p\} = \begin{Bmatrix} X \\ Y \\ Z \end{Bmatrix} = \sum_{i=1}^3 N_i \begin{Bmatrix} X_i \\ Y_i \\ Z_i \end{Bmatrix} \quad (2.110)$$

Dans cette équation, les N_i représentent évidemment les fonctions de transformation géométrique définies à la section 2.1.4.1 (équation 2.14). Aussi, les vecteurs de base $\{a_1\}$ et $\{a_2\}$ sont :

$$\{a_1\} = \{x_{p,\xi}\} = \begin{Bmatrix} X_2 - X_1 \\ Y_2 - Y_1 \\ Z_2 - Z_1 \end{Bmatrix} \quad \{a_2\} = \{x_{p,\eta}\} = \begin{Bmatrix} X_3 - X_1 \\ Y_3 - Y_1 \\ Z_3 - Z_1 \end{Bmatrix} \quad (2.111)$$

À partir de ces résultats, il est possible de définir un repère $[Q] = [\mathbf{t}_1 \mid \mathbf{t}_2 \mid \mathbf{n}]$ avec :

$$\{n\} = \begin{Bmatrix} n_X \\ n_Y \\ n_Z \end{Bmatrix} = \frac{1}{2A} \begin{Bmatrix} Y_{21}Z_{31} - Y_{31}Z_{21} \\ Z_{21}X_{31} - Z_{31}X_{21} \\ X_{21}Y_{31} - X_{31}Y_{21} \end{Bmatrix} \quad (2.112)$$

$$\langle t_1 \rangle = \left\langle n_Z + \frac{n_Y^2}{a} \quad -\frac{n_X n_Y}{a} \quad -n_X \right\rangle \quad (2.113)$$

$$\langle t_2 \rangle = \left\langle -\frac{n_X n_Y}{a} \quad n_Z + \frac{n_X^2}{a} \quad -n_Y \right\rangle \quad (2.114)$$

avec

$$a = 1 + n_z \quad X_{ij} = X_i - X_j \quad A = \frac{|\mathbf{a}_1 \wedge \mathbf{a}_2|}{2} \quad (2.115)$$

Si $a = 0$ dans les équations 2.113 et 2.114, on a :

$$\langle n \rangle = \langle 0 \ 0 \ -1 \rangle \quad \langle t_1 \rangle = \langle 1 \ 0 \ 0 \rangle \quad \langle t_2 \rangle = \langle 0 \ -1 \ 0 \rangle \quad (2.116)$$

Avec ces relations, les coordonnées des nœuds 1, 2 et 3 dans le repère local sont :

$$\begin{aligned} \begin{Bmatrix} x_1 \\ y_1 \end{Bmatrix} &= \begin{Bmatrix} 0 \\ 0 \end{Bmatrix} & \begin{Bmatrix} x_2 \\ y_2 \end{Bmatrix} &= [\mathbf{t}_1 \ \mathbf{t}_2]^T \begin{Bmatrix} X_{21} \\ Y_{21} \\ Z_{21} \end{Bmatrix} \\ \\ \begin{Bmatrix} x_3 \\ y_3 \end{Bmatrix} &= [\mathbf{t}_1 \ \mathbf{t}_2]^T \begin{Bmatrix} X_{31} \\ Y_{31} \\ Z_{31} \end{Bmatrix} \end{aligned}$$

2.3.2 Expression de W_{int}^e dans le repère local

La forme discrétisé de W_{int}^e relative à un élément de membrane et de flexion est :

$$W_{\text{int}}^e = \langle u_n^* \rangle ([k] \{u_n\} + \{f_\sigma\}) \quad (2.117)$$

avec

$$\langle u_n \rangle_{\text{loc}} = \langle \dots | u_i \ v_i \ w_i \ \theta_{zi} \ \theta_{yi} \ \theta_{xi} | i = 1, 2, 3 \dots \rangle \quad (2.118)$$

$$[k] = [k_m] + [k_{mf}] + [k_f] \quad (2.119)$$

$$[k_m] = \int_{A^e} [B_m]^T [H_m] [B_m] dA \quad (2.120)$$

$$[k_{mf}] = \int_{A^e} ([B_m]^T [H_m] [B_f] + [B_f]^T [H_{mf}] [B_m]) dA \quad (2.121)$$

$$[k_f] = \int_{A^e} [B_f]^T [H_f] [B_f] dA \quad (2.122)$$

$$\{f_\sigma\} = \int_{A^e} ([B_m]^T \{N_0\} + [B_f]^T \{M_0\}) dA \quad (2.123)$$

La matrice $[k_m]$ relative à l'élément CST est définie à la section 2.1. La matrice $[k_f]$ relative à l'élément DST est définie à la section 2.2. Dans le cas d'une plaque définie à partir d'un matériau isotrope ou d'un stratifié symétrique, la matrice $[H_{mf}]$ est nulle et par conséquent, la matrice $[k_{mf}]$ est nulle. Le vecteur $\{f_\sigma\}$ est relatif aux efforts dus aux contraintes initiales. Toutes ces matrices sont intégrées numériquement à l'aide de trois points de Hammer.

Pour un stratifié non-symétrique, la matrice $[H_{mf}]$ est non nulle et la matrice de rigidité $[k_{mf}]$ doit être calculée. Dans ce cas, la matrice $[B_m]$ doit être calculée à partir de l'équation 2.16 et la matrice $[B_f]$ à partir de l'équation 2.97. Par contre, il est important de noter que pour un composite non-symétrique, la méthode de calcul de la matrice H_c décrite à la section A.3.1 n'est pas valide.

2.3.3 Rigidité fictive pour θ_z

Le lecteur est en mesure de remarquer que les relations cinématiques données jusqu'à maintenant ne font pas intervenir les rotations autour de la normale θ_z . Ainsi, tous les termes de la matrice $[k]$ (équation 2.119) associés aux variables nodales θ_{zi} sont nuls. Par conséquent, si tous les éléments connectés à un nœud i sont coplanaires, la matrice de rigidité assemblée sera singulière.

Afin d'éviter cette difficulté numérique, il est possible d'introduire une matrice de rigidité fictive de telle que :

$$W_{\text{int}}^e = W_{\text{int}}^e + W_{\theta_z}^e \quad (2.124)$$

où W_{int}^e est défini par 2.117 et

$$W_{\theta_z}^e = \int_{A^e} \alpha E \frac{h^3}{12} (\theta_{z,x}^* \theta_{z,x} + \theta_{z,y}^* \theta_{z,y}) dA \quad (2.125)$$

Dans la dernière équation, α est un coefficient sans dimension petit par rapport à l'unité. La valeur de α dépend de la précision de l'ordinateur. Elle doit être suffisamment petite pour limiter l'influence des rigidités fictives sur le résultat et suffisamment grande pour éviter les singularités possibles. La valeur de α choisie dans notre solveur élément finis est prise égale à 10^{-4} (voir référence [8, §6.4.2.3]).

Nous obtenons finalement :

$$W_{\theta_z}^e = \langle \theta_{zn}^* \rangle [k_{\theta_z}] \{ \theta_{zn} \} \quad (2.126)$$

$$[k_{\theta_z}] = \alpha E \frac{h^3}{24A} \begin{bmatrix} y_{32}^2 + x_{32}^2 & -y_{32}y_3 - x_3x_{32} & y_{32}y_2 + x_{32}x_2 \\ & y_3^2 + x_3^2 & -y_3y_2 - x_3x_2 \\ \text{sym.} & & y_2^2 + x_2^2 \end{bmatrix} \quad (2.127)$$

Pour un stratifié, la valeur du module d'Young n'est pas clairement définie. Dans ce cas, le coefficient E peut être pris égale à un module de rigidité équivalent correspondant à E_1 .

2.3.4 Expression de W^e dans le repère global

Les déplacements et les rotations dans le repère globale X, Y, Z sont $U, V, W, \theta_X, \theta_Y, \theta_Z$. Ces déplacements et ces rotations sont reliés aux quantités locales de la façon suivante :

$$\begin{Bmatrix} u \\ v \\ w \end{Bmatrix} = [Q]^T \begin{Bmatrix} U \\ V \\ W \end{Bmatrix} \quad \begin{Bmatrix} \theta_x \\ \theta_y \\ \theta_z \end{Bmatrix} = [Q]^T \begin{Bmatrix} \theta_X \\ \theta_Y \\ \theta_Z \end{Bmatrix} \quad (2.128)$$

Les composantes de $[Q]^T$ sont données explicitement par les équations 2.112, 2.113 et 2.114. Les variables nodales de l'élément DST18 dans le repère global sont :

$$\langle u_n \rangle_{\text{glob}} = \langle \dots \mid U_i \ V_i \ W_i \ \theta_{Xi} \ \theta_{Yi} \ \theta_{Zi} \mid i = 1, 2, 3 \ \dots \rangle \quad (2.129)$$

À l'aide des équations 2.118, 2.128 et 2.129, il est possible d'obtenir les relations suivantes :

$$\{u_n\}_{\text{loc}} = [T]\{u_n\}_{\text{glob}} \quad \{u_n^*\}_{\text{loc}} = [T]\{u_n^*\}_{\text{glob}} \quad (2.130)$$

avec

$$[T]_{18 \times 18} = \begin{bmatrix} [A] & [0] & [0] \\ [0] & [A] & [0] \\ [0] & [0] & [A] \end{bmatrix} \quad [A]_{6 \times 6} = \begin{bmatrix} [Q]^T & [0] \\ [0] & [Q]^T \end{bmatrix} \quad (2.131)$$

À partir des relations définies jusqu'à maintenant, il nous est possible de définir les matrices et vecteurs élémentaires dans le repère global :

$$[k]_{\text{glob}} = [T]^T [k]_{\text{loc}} [T] \quad \{f_\sigma\}_{\text{glob}} = [T]^T \{f_\sigma\}_{\text{loc}} \quad (2.132)$$

Étant donné que les matrices élémentaires seront maintenant exprimées par rapport à un même système d'axe global, il sera possible de les assembler et ainsi obtenir la matrice de rigidité globale du système d'équations. Après avoir résolu ce système d'équations linéaires, les variables nodales globales pourront être *tournées* à l'aide de l'équation 2.130 de façon à obtenir les variables nodales dans le système local de chacun des éléments. C'est avec ces variables nodales *locales* que le calcul des quantités d'intérêt comme les contraintes et les déformations pourra être effectué.

Chapitre 3

Estimateur d'erreur et adaptation de maillage

L'adaptation de maillage est basée sur deux ingrédients principaux. Premièrement, l'erreur doit être estimée. Ensuite, l'estimé est utilisé pour modifier le maillage de façon à équidistribuer l'estimation de l'erreur, c'est-à-dire à la rendre égale sur toutes les arêtes du maillage. Dans ce chapitre, nous présentons une brève description des bases théoriques qui justifient l'utilisation de l'estimateur d'erreur proposé et validé pour des problèmes d'aérodynamique à la référence [12]. Ensuite, nous décrivons les stratégies d'adaptation de maillage qui ont été programmées dans la librairie d'optimisation de maillages "Libom".

3.1 Description de l'estimateur d'erreur utilisé

Soit u_{ex} la solution exacte d'un problème aux dérivées partielles et u_h une solution approximative obtenue par éléments finis de ce même problème. L'erreur entre la solution exacte et la solution numérique approximative est définie comme étant $\|u_{ex} - u_h\|$. Normalement, un bon processus d'adaptation devrait faire converger u_h vers u_{ex} et ainsi faire tendre la norme de l'erreur vers zéro.

Le but d'un processus d'adaptation de maillage est donc de construire un maillage (ou de le modifier s'il existe) de telle sorte que l'erreur soit distribué uniformément. La solution de ce problème d'*optimisation* est assez simple. Il suffit d'utiliser un nombre infini de nœuds. Cette solution n'est par contre pas réalisable en pratique. Il est donc plus logique de définir le but d'un processus d'adaptation comme étant de trouver le maillage pour lequel l'erreur sur chacune des arêtes est la même. Un maillage respectant ce critère sera un maillage *adapté*. Modifier ou construire un maillage qui minimise l'erreur suppose que l'erreur puisse être estimée.

Habashi *et al.* [12] ont développé un estimateur d'erreur basé sur les dérivées secondes de la solution recherchée. Cet estimateur est indépendant de la physique du problème et a déjà donné d'excellents résultats en dynamique des fluides. L'estimateur correspond à l'erreur d'interpolation équivalente à la longueur d'une arête dans le repère d'une métrique calculée à partir de la matrice des dérivées secondes (*le Hessien*) de la solution approximée. En plus de nous indiquer l'erreur à un endroit donné du domaine, cet estimateur nous donne des informations *directionnelles* sur la variation de la solution. Il permet donc de modifier et la dimension et la forme des éléments.

L'estimateur d'erreur repose sur une approximation linéaire des variables nodales. Nous avons vu, au chapitre précédent, que l'élément de coque plate qui est utilisé est composé d'une partie membrane et d'une partie flexion. La partie membrane utilise une approximation *linéaire* des composantes u et v du déplacement alors que la partie flexion utilise une approximation *quadratique* des composantes β_x et β_y de la rotation de la normale. Une approximation quadratique des rotations est compatible avec une approximation *cubique* de continuité C^1 du déplacement transverse w . Ceci a pour conséquence que l'estimateur d'erreur utilisé est non optimal dans le sens où l'erreur est probablement plus petite que l'estimé que nous utilisons.

Soit $P_h u_{ex}$ une interpolation linéaire de la solution exacte u_{ex} . L'*erreur d'interpolation* entre la solution exacte u_{ex} et la solution interpolée linéairement $P_h u_{ex}$ est définie comme étant $\|u_{ex} - P_h u_{ex}\|$. L'objectif est donc d'utiliser l'erreur d'interpolation comme indicateur de l'erreur réelle $\|u_{ex} - u_h\|$. L'erreur sur la solution obtenue par éléments finis est bornée par l'erreur d'interpolation [12] :

$$\|u_{ex} - u_h\| \leq C \|u_{ex} - P_h u_{ex}\| \quad (3.1)$$

Ainsi, si le processus d'adaptation diminue l'erreur d'interpolation $\|u_{ex} - P_h u_{ex}\|$, l'erreur sur la solution éléments finis sera aussi diminuée. Par conséquent, l'erreur sur la solution obtenue par éléments finis sera contrôlée en utilisant un processus d'adaptation qui régit l'erreur d'interpolation.

3.1.1 Évaluation de l'erreur d'interpolation

Le but de cette section est de définir l'erreur d'interpolation pour un élément 2D linéaire à trois nœuds. L'erreur en tout point \mathbf{x} d'un élément réel est définie par :

$$e(\mathbf{x}) = \|u_{ex}(\mathbf{x}) - u_h(\mathbf{x})\| \quad (3.2)$$

L'erreur au point (ξ, η) de l'élément de référence correspondant au point \mathbf{x} de l'élément réel est :

$$e(\xi, \eta) = e(\mathbf{x}) = \|u_{ex}(\xi, \eta) - u_h(\xi, \eta)\| \quad (3.3)$$

Pour pouvoir évaluer l'erreur $e(\xi, \eta)$, nous développons en série de Taylor la solution exacte u_{ex} au voisinage du point (ξ, η) . Pour un problème bidimensionnel, on obtient [5, §1.7] :

$$\begin{aligned} u_{ex}(\xi + h, \eta + k) = & u_{ex}(\xi, \eta) + \left\{ \frac{\partial u_{ex}(\xi, \eta)}{\partial \xi} h \Big|_{\xi, \eta} + \frac{\partial u_{ex}(\xi, \eta)}{\partial \eta} k \Big|_{\xi, \eta} \right\} \\ & + \frac{1}{2!} \left\{ \frac{\partial^2 u_{ex}(\xi, \eta)}{\partial \xi^2} h^2 \Big|_{\xi, \eta} + 2 \frac{\partial^2 u_{ex}(\xi, \eta)}{\partial \xi \partial \eta} kh \Big|_{\xi, \eta} + \frac{\partial^2 u_{ex}(\xi, \eta)}{\partial \eta^2} k^2 \Big|_{\xi, \eta} \right\} \\ & + \frac{1}{3!} \{ \dots \} + \dots + \frac{1}{n!} \{ \dots \} + R_n \end{aligned} \quad (3.4)$$

Pour un élément possédant 3 nœuds d'interpolation de coordonnées ξ_i et η_i ($i = 1, 2, 3$), l'équation 3.4 peut être modifiée en choisissant $h = \xi_i - \xi$ et $k = \eta_i - \eta$:

$$\begin{aligned}
 u_{ez}(\xi_i, \eta_i) = & u_{ez}(\xi, \eta) + \left\{ (\xi_i - \xi) \frac{\partial u_{ez}}{\partial \xi} \Big|_{\xi, \eta} + (\eta_i - \eta) \frac{\partial u_{ez}}{\partial \eta} \Big|_{\xi, \eta} \right\} \\
 & + \frac{1}{2!} \left\{ (\xi_i - \xi)^2 \frac{\partial^2 u_{ez}}{\partial \xi^2} \Big|_{\xi, \eta} + 2(\eta_i - \eta)(\xi_i - \xi) \frac{\partial^2 u_{ez}}{\partial \xi \partial \eta} \Big|_{\xi, \eta} + (\eta_i - \eta)^2 \frac{\partial^2 u_{ez}}{\partial \eta^2} \Big|_{\xi, \eta} \right\} \\
 & + \frac{1}{3!} \{ \dots \} + \dots + \frac{1}{n!} \{ \dots \} + R_n
 \end{aligned} \tag{3.5}$$

En reportant l'expression précédente de $u_{ez}(\xi_i, \eta_i)$ correspondant à la solution exacte du problème à chaque nœud i dans la fonction approchée par éléments finis, il est possible de relier la valeur approchée $u(\xi, \eta)$ à la valeur exacte $u_{ez}(\xi, \eta)$:

$$u(\xi, \eta) = \sum_i N_i u(\xi_i, \eta_i) \tag{3.6}$$

on a

$$\begin{aligned}
 u(\xi, \eta) = & \left(\sum_i N_i \right) u_{ez}(\xi, \eta) + \left(\sum_i N_i (\xi_i - \xi) \right) \frac{\partial u_{ez}}{\partial \xi} \Big|_{\xi, \eta} \\
 & + \left(\sum_i N_i (\eta_i - \eta) \right) \frac{\partial u_{ez}}{\partial \eta} \Big|_{\xi, \eta} + \frac{1}{2} \left(\sum_i N_i (\xi_i - \xi)^2 \right) \frac{\partial^2 u_{ez}}{\partial \xi^2} \Big|_{\xi, \eta} \\
 & + \left(\sum_i N_i (\eta_i - \eta) (\xi_i - \xi) \right) \frac{\partial^2 u_{ez}}{\partial \xi \partial \eta} \Big|_{\xi, \eta} + \frac{1}{2} \left(\sum_i N_i (\eta_i - \eta)^2 \right) \frac{\partial^2 u_{ez}}{\partial \eta^2} \Big|_{\xi, \eta} \\
 & + \frac{1}{3!} \{ \dots \} + \dots + \frac{1}{n!} \{ \dots \} + R_n
 \end{aligned} \tag{3.7}$$

puisque pour une base polynomiale complète de degré 1 on a :

$$\begin{aligned}
 \sum_i N_i &= 1 \\
 \sum_i N_i (\xi_i - \xi) &= 0 \\
 \sum_i N_i (\eta_i - \eta) &= 0
 \end{aligned} \tag{3.8}$$

L'équation 3.3 qui donne l'expression de l'erreur, devient en négligeant les termes d'ordre supérieur à 2 :

$$\begin{aligned}
 e(\xi, \eta) = \|u(\xi, \eta) - u_{ex}(\xi, \eta)\| &= \frac{1}{2} \sum_i \left(N_i(\xi_i - \xi)^2 \frac{\partial^2 u_{ex}}{\partial \xi^2} \Big|_{\xi, \eta} \right) \\
 &+ \sum_i \left(N_i(\xi_i - \xi)(\eta_i - \eta) \frac{\partial^2 u_{ex}}{\partial \xi \partial \eta} \Big|_{\xi, \eta} \right) + \frac{1}{2} \sum_i \left(N_i(\eta_i - \eta)^2 \frac{\partial^2 u_{ex}}{\partial \eta^2} \Big|_{\xi, \eta} \right) \quad (3.9)
 \end{aligned}$$

ou sous forme matricielle :

$$e(\xi, \eta) = \frac{1}{2} \sum_i \left(\langle N_i(\xi_i - \xi) \quad N_i(\eta_i - \eta) \rangle [H_\xi] \begin{Bmatrix} (\xi_i - \xi) \\ (\eta_i - \eta) \end{Bmatrix} \right) \quad (3.10)$$

avec

$$[H_\xi] = \begin{bmatrix} \frac{\partial^2 u_{ex}(\xi, \eta)}{\partial \xi^2} \Big|_{\xi, \eta} & \frac{\partial^2 u_{ex}(\xi, \eta)}{\partial \xi \partial \eta} \Big|_{\xi, \eta} \\ \frac{\partial^2 u_{ex}(\xi, \eta)}{\partial \xi \partial \eta} \Big|_{\xi, \eta} & \frac{\partial^2 u_{ex}(\xi, \eta)}{\partial \eta^2} \Big|_{\xi, \eta} \end{bmatrix} \quad (3.11)$$

Cette expression fait intervenir les dérivées secondes en ξ et η de la fonction u_{ex} . Pour une transformation géométrique linéaire, ces dérivées sont liées aux dérivées en x et y calculées sur l'élément réel par la relation [5, §1.5.3] :

$$\begin{Bmatrix} \frac{\partial^2}{\partial \xi^2} \\ \frac{\partial^2}{\partial \eta^2} \\ \frac{\partial^2}{\partial \xi \partial \eta} \end{Bmatrix} = \begin{bmatrix} J_{11}^2 & J_{12}^2 & 2J_{11}J_{12} \\ J_{21}^2 & J_{22}^2 & 2J_{21}J_{22} \\ J_{11}J_{21} & J_{12}J_{22} & J_{11}J_{22} + J_{12}J_{21} \end{bmatrix} \begin{Bmatrix} \frac{\partial^2}{\partial x^2} \\ \frac{\partial^2}{\partial y^2} \\ \frac{\partial^2}{\partial x \partial y} \end{Bmatrix} \quad (3.12)$$

Les différents termes de la dernière équation sont les termes de la matrice jacobienne. Puisqu'une transformation géométrique linéaire est utilisée, les termes de la matrice jacobienne sont constants sur un élément donné. À partir de l'équation 3.11 et 3.12, on voit que l'erreur d'interpolation dépend du Hessien $[H]$ et de la taille des éléments. Pour équidistribuer l'erreur, on doit donc augmenter la taille des éléments là où les dérivées secondes de u_{ex} sont petites et vice-versa. La prochaine section présente la façon par lequel le Hessien peut être évalué.

3.1.2 Évaluation du Hessien

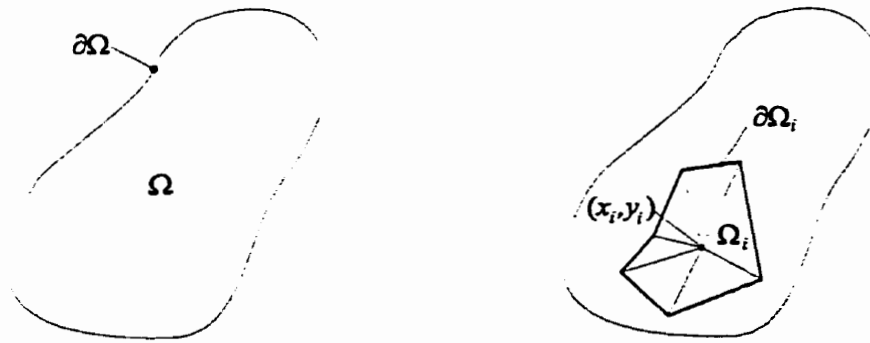
Pour évaluer l'erreur d'interpolation, nous devons évaluer le Hessien de la solution. Il est important de se rappeler que la solution est l'inconnue du problème. Par

conséquent, le Hessien de la solution est aussi inconnu. Pour contourner ce problème, le Hessien est approché en le calculant à partir de la solution numérique obtenue par éléments finis ($[H(u_{ex})] \simeq [H(u_h)]$). Ainsi, lorsque la solution numérique convergera vers la solution exacte, $[H(u_h)]$ convergera vers $[H(u_{ex})]$ [12, §2.5].

Cependant, puisque les fonctions d'interpolation de la solution numérique u_h sont linéaires, les dérivées secondes calculées à partir de celles-ci seront donc nulles. Il existe plusieurs moyens dans la littérature [12] pour surmonter cette difficulté. La méthode retenue dans "Libom" pour calculer les dérivées secondes consiste à utiliser une forme intégrale faible. Par exemple pour le terme $u_{,xy}$, on peut écrire :

$$\int_{\Omega} \frac{\partial^2 u}{\partial x \partial y} \phi \, dA = - \int_{\Omega} \frac{\partial u}{\partial x} \frac{\partial \phi}{\partial y} \, dA + \oint_{\partial \Omega} \frac{\partial u}{\partial x} \phi n_y \, dS \quad (3.13)$$

Cette formule peut être employée pour des interpolations linéaires car le membre de droite ne contient que des dérivées d'ordre 1. ϕ est une fonction de pondération qui satisfait les conditions aux limites et qui s'annule sur la frontière du domaine $\partial \Omega$ (voir figure 3.1(a)).


 (a) Domaine Ω et frontière $\partial \Omega$

 (b) Domaine Ω_i et frontière $\partial \Omega_i$

FIGURE 3.1: Description des domaines et de leur frontière

À l'aide de l'équation 3.13, nous obtenons une approximation linéaire $D_{xy}(u_h)$ de $\frac{\partial^2 u_h}{\partial x \partial y}$:

$$\int_{\Omega} D_{xy} u_h \phi \, dA = - \int_{\Omega} \frac{\partial u_h}{\partial x} \frac{\partial \phi}{\partial y} \, dA + \oint_{\partial \Omega} \frac{\partial u_h}{\partial x} \phi n_y \, dS \quad (3.14)$$

À partir de cette équation, les dérivées secondes $D_{xy}(u_h)$ à chaque nœud peuvent être approchées de la façon suivante :

$$\frac{\partial^2 u_h}{\partial x \partial y}(x_i, y_i) \approx \frac{-\int_{\Omega_i} \frac{\partial u_h}{\partial x} \frac{\partial \phi_i}{\partial y} dA + \oint_{\partial \Omega_i} \frac{\partial u_h}{\partial x} \phi_i n_y dS}{\int_{\Omega_i} \phi_i dA} \quad (3.15)$$

où ϕ_i est la fonction de pondération évaluée au nœud de coordonnées (x_i, y_i) et Ω_i représente un domaine composé de tous les éléments adjacents à ce nœud (voir figure 3.1(b)). Ainsi, il est possible d'évaluer les dérivées secondes à un nœud quelconque du maillage à partir des informations fournies par les éléments adjacents seulement. Des expressions similaires sont utilisées pour estimer $\frac{\partial^2 u_h}{\partial x^2}$ et $\frac{\partial^2 u_h}{\partial y^2}$.

Pour les nœuds intérieurs, les fonctions de pondération ϕ_i s'annule sur les bords du domaine Ω_i . Par conséquent, le deuxième terme du numérateur de l'équation 3.15 s'annule et il n'est donc pas nécessaire de l'évaluer. Par contre, pour un nœud sur une frontière, ce terme devrait être évalué. Dans la librairie d'adaptation de maillage "Libom", l'intégrale de contour n'est jamais calculée. La stratégie utilisée est plutôt d'extrapoler la valeur des dérivées secondes à un nœud de la frontière en utilisant la valeur obtenue au nœud interne le plus près.

3.1.3 Évaluation de l'erreur sur une arête

Nous avons vu, à la section précédente, comment mesurer l'erreur à partir du Hessien de la solution à un point donné. Nous allons maintenant définir le calcul de l'erreur sur une arête. Pour une arête $\gamma = [\mathbf{x}_0, \mathbf{x}_1]$ définie par ses nœuds d'extrémités de coordonnées \mathbf{x}_0 et \mathbf{x}_1 , l'erreur nous est donnée par :

$$e(\gamma) = \int_{\mathbf{x}_0}^{\mathbf{x}_1} \sqrt{(\mathbf{x}_1 - \mathbf{x}_0)^T \mathbf{M}(s) (\mathbf{x}_1 - \mathbf{x}_0)} ds \quad (3.16)$$

Dans cette équation, on remarque que le Hessien n'est pas utilisé mais plutôt une métrique \mathbf{M} dérivée de celui-ci. Il n'est pas garanti que le Hessien soit défini positif. Le terme $(\mathbf{x}_1 - \mathbf{x}_0)^T [H(s)] (\mathbf{x}_1 - \mathbf{x}_0)$ pourrait donc être négatif. Afin d'éviter ceci, le Hessien qui peut être exprimé sous la forme suivante :

$$\mathbf{H}(x, y) = \begin{bmatrix} \frac{\partial^2 u_h(x, y)}{\partial x^2} & \frac{\partial^2 u_h(x, y)}{\partial x \partial y} \\ \frac{\partial^2 u_h(x, y)}{\partial x \partial y} & \frac{\partial^2 u_h(x, y)}{\partial y^2} \end{bmatrix} = \mathbf{Q}(x, y) \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \mathbf{Q}(x, y)^T \quad (3.17)$$

est rendu défini positif comme suit :

$$\mathbf{M}(x, y) = \mathbf{Q}(x, y) \begin{bmatrix} |\lambda_1| & 0 \\ 0 & |\lambda_2| \end{bmatrix} \mathbf{Q}(x, y)^T \quad (3.18)$$

où λ_1 et λ_2 sont les valeurs propres du Hessien et \mathbf{Q} est la matrice de ses vecteurs propres. Les quantités λ_1 et λ_2 sont directement proportionnelles aux dimensions de l'élément. L'erreur d'interpolation sur une arête basée sur le Hessien est la longueur de cette arête dans le repère de la métrique \mathbf{M} . Le lecteur qui voudrait avoir plus d'informations sur ce sujet peut consulter la référence [13].

Le calcul de l'erreur d'interpolation sur une arête basée sur les équations 3.16 et 3.18 doit être réalisé avec précaution. Une vérification doit être faite afin que les valeurs propres λ_1 et λ_2 ne soient pas à l'extérieur d'un interval $[\lambda_{\min}, \lambda_{\max}]$ et cela afin d'éviter d'obtenir des éléments trop étirés. Les quantités λ_{\min} et λ_{\max} sont en fait les longueurs euclidiennes minimum et maximum d'une arête. L'utilisateur a généralement une bonne idée de ces quantités et fixer les valeurs de λ_{\min} et λ_{\max} ne cause donc pas de difficultés.

3.1.4 Remarques

Il est important de rappeler que les équations développées précédemment ne sont valides que pour des éléments basés sur un interpolation linéaire. L'élément présenté au chapitre 2 est composé d'un élément linéaire en membrane et d'un élément cubique[†] de continuité C^1 en flexion. Pour une plaque fabriquée d'un matériau qui n'entraîne pas de couplage membrane-flexion, le déplacement résultant, qui est utilisé comme critère d'adaptation, contient donc des composantes linéaires (u et v) et une composante cubique (w).

[†]Voir référence [7], page 338

Afin de pouvoir utiliser les résultats développés jusqu'à maintenant, nous allons considérer que le déplacement résultant varie linéairement. Ceci signifie que pour les problèmes pour lesquels w est dominant par rapport à u et v , le nombre d'élément obtenu après l'adaptation du maillage sera supérieur à celui qui serait normalement requis puisque la solution sera dominée par l'interpolation cubique. Dans ce sens, l'estimateur qui est proposé est donc non optimal.

3.2 Stratégie d'adaptation de maillage

Ayant en main des outils nous permettant d'estimer l'erreur sur une arête, la prochaine étape est de définir un maillage qui permettra de l'équidistribuer. Par définition, l'erreur est équidistribuée sur un maillage lorsque l'erreur est la même sur chacune des arêtes. Cette stratégie permet d'obtenir des maillages anisotropes puisque l'erreur sur une arête permet de contrôler individuellement la longueur de chacune des arêtes. Pour y arriver, les techniques suivantes sont utilisées [12] :

- raffinement et déraffinement du maillage (*h-method*);
- changement dans la connectivité des nœuds (pivotement des arêtes);
- relocalisation des nœuds (*r-method*).

De façon générale, la procédure à suivre lors d'un processus d'adaptation de maillage complet est donnée par l'algorithme présenté à la figure 3.2.

Pour démarrer, le processus d'adaptation exige un maillage de départ, une solution de départ et un choix du niveau d'erreur. Dans la phase *initialisation du maillage de fond*, la métrique M est calculée à chaque nœud. Ainsi, à partir de ces valeurs aux nœuds, on définit un champs linéaire de la métrique sur le maillage de départ (maillage de fond). Avec ce maillage de fond, il sera possible de connaître la valeur de la métrique en tout point du domaine. L'évaluation de l'erreur à partir de l'équation 3.16 qui est faite à l'aide d'une intégration numérique ne cause donc plus un problème puisque à chaque point d'intégration, la métrique pourra être connue. Avec l'aide de l'erreur calculée sur chaque arête, le maillage *adapté* est construit. Sur

1. Calculer un maillage et une solution de départ.
2. Choix d'une longueur optimale des arêtes dans la métrique (choix du niveau d'erreur).
3. Faire jusqu'à convergence
 - Initialisation du maillage de fond
 - Calcul de l'erreur sur chaque arête.
 - À partir du maillage de fond, construction du maillage adapté.
 - Calcul d'une nouvelle solution sur le maillage adapté.
4. Fin.

FIGURE 3.2: Description du processus d'adaptation

ce maillage adapté, une nouvelle solution est calculée et le maillage adapté devient le nouveau maillage de fond pour la prochaine itération et le processus recommence jusqu'à la convergence.

La façon dont est modifié le maillage à partir de l'estimateur d'erreur n'a pas été décrite en détails ici mais le lecteur peut consulter Habashi et *al.* [12, 13] pour plus d'explications à ce sujet.

Chapitre 4

Résultats de l'adaptation de maillage

Le but de ce chapitre est de démontrer que la stratégie d'adaptation de maillages et de calcul de l'erreur décrite au chapitre précédent peut être efficacement utilisée dans le cas de plaques fléchies. Bien que le résolveur éléments finis présenté au chapitre 2 puisse être utilisé pour l'étude de coques, la librairie d'adaptation "Libom" ne fonctionne que sur des domaines 2D et c'est pourquoi seules des plaques 2D seront étudiées dans ce chapitre.

Afin de vérifier que "Libom" adapte correctement dans un contexte de plaques fléchies, cinq cas sont étudiés :

- une plaque en porte-à-faux chargée en membrane ;
- une plaque en porte-à-faux chargée en flexion ;
- une plaque simplement supportée chargée en flexion ;
- une plaque carrée simplement supportée chargée en flexion ;
- une plaque trouée simplement supportée chargée en flexion.

Notons que pour les quatrième et cinquième cas, la plaque est fabriquée à l'aide d'un matériau orthotrope alors que pour tous les autres cas, le matériau utilisé est isotrope.

Les cas relativement simples étudiés ici permettront de comparer les solutions obtenues aux solutions exactes. Il ne s'agit donc pas de problèmes ayant de grandes significations pratiques mais le but est de démontrer que la stratégie d'adaptation de maillages présentée au chapitre précédent donne de bons résultats en analyse des structures.

Pour chacun des cas qui suivront, nous présenterons la géométrie et les propriétés du modèle à l'étude. Ensuite, nous présenterons la solution théorique du problème analysé. Nous avons choisi d'utiliser comme critère d'adaptation l'erreur mesurée à partir du déplacement résultant calculé sur le plan moyen. L'utilisation du déplacement résultant pour le calcul de l'erreur permet plus facilement de tenir compte des différentes composantes du déplacement dans le cas d'une plaque fléchie. De plus, peu importe la formulation de l'élément de plaque utilisée, le déplacement résultant est toujours quelque chose qui est disponible.

À partir de la solution théorique, il est possible de calculer l'erreur existant entre la solution éléments finis notée \mathbf{u}_h et la solution théorique notée \mathbf{u}_{ex} pour les différents maillages obtenus. Ainsi, il sera possible de mesurer dans quelle mesure le processus d'adaptation permet de diminuer l'erreur et de faire converger la solution éléments finis vers la solution théorique.

Afin de bien apprécier l'évolution de l'erreur sur chacun des maillages au cours d'un processus d'adaptation, différentes quantités seront calculées. À l'aide de la solution théorique et la solution obtenue par éléments finis, les calculs suivant sont réalisés :

$$\psi_i = \frac{\int_{\gamma_i} \|\mathbf{u}_h\| d\gamma_i}{\int_{\gamma_i} \|\mathbf{u}_{ex}\| d\gamma_i} \quad (4.1)$$

$$\text{erreur} = \frac{1}{n} \sum_{i=1}^n \left(\int_{\gamma_i} \|\mathbf{u}_h\| d\gamma_i - \int_{\gamma_i} \|\mathbf{u}_{ex}\| d\gamma_i \right) \quad (4.2)$$

Dans les équations 4.1 et 4.2, les quantités $\|\mathbf{u}_h\|$ et $\|\mathbf{u}_{ex}\|$ représentent le déplacement résultant pour la solution éléments finis et la solution exacte respectivement. Les déplacements résultants sont calculés à partir de la norme L_2 du vecteur des

déplacements. Aussi, γ_i représente le domaine occupé par l'arête i et n le nombre total d'arête. Normalement, la moyenne des ψ_i calculés sur toutes les arêtes du maillage devrait tendre vers l'unité. Ceci implique que l'erreur tend vers zéro. Dans les sections qui suivront, l'erreur sera celle calculée à l'aide de l'équation 4.2. Les unités représentant l'erreur seront donc des unités du déplacement au carré.

À partir des démonstrations faites au chapitre précédent, le niveau d'erreur sera choisi en imposant la longueur optimale d'une arête dans le repère de la métrique. Ainsi, en faisant ce choix, nous pourrons imposer le niveau d'erreur désiré. Puisque la longueur optimale dans la métrique correspond à l'erreur intégrée sur une arête, les unités représentant la longueur optimale dans la métrique sont aussi des unités du déplacement au carré.

4.1 Cas 1 : Poutre en porte-à-faux cisailée

La poutre étudiée (voir figure 4.1) a une épaisseur de 1 mm et est sollicitée par une force de cisaillement uniformément répartie appliquée à son extrémité libre. L'intensité de la charge est de 1 N/mm répartie sur une hauteur h de 10 mm pour une charge totale P de 10 N . Le module d'Young E est de 200000 MPa et le coefficient de Poisson ν est de 0.3

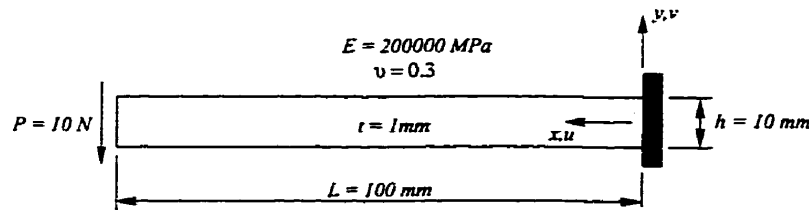


FIGURE 4.1: Description du modèle

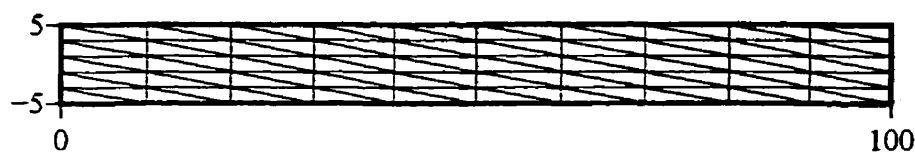
Pour une poutre en porte-à-faux soumise à une charge concentrée, les déplacements u et v mesurés le long de x et y sont donnés par [30] :

$$\begin{aligned} u(x, y) &= -\frac{1}{2} \frac{Px^2y}{EI} - \frac{1}{6} \frac{\nu Py^3}{EI} + \frac{1}{6} \frac{Py^3}{GI} + \frac{1}{2} \frac{PL^2y}{EI} - \frac{1}{8} \frac{Ph^2y}{GI} \\ v(x, y) &= \frac{1}{2} \frac{\nu Pxy^2}{EI} + \frac{1}{6} \frac{Px^3}{EI} - \frac{1}{2} \frac{PL^2x}{EI} + \frac{1}{3} \frac{PL^3}{EI} \end{aligned} \quad (4.3)$$

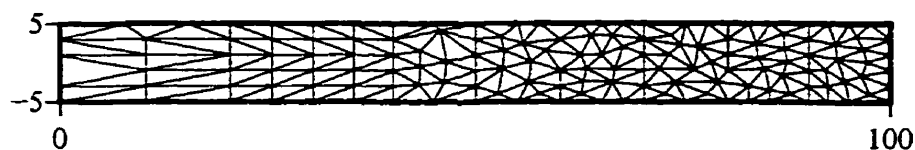
où I représente l'inertie selon l'axe z et G le module de cisaillement. La flèche au bout du porte-à-faux calculée à mi-hauteur $v(100,0)$ est 0.02 mm . La sous-section suivante présente les résultats obtenus des processus d'adaptation ainsi que les erreurs calculées pour chacun des maillages obtenus.

4.1.1 Résultats de l'adaptation de maillage

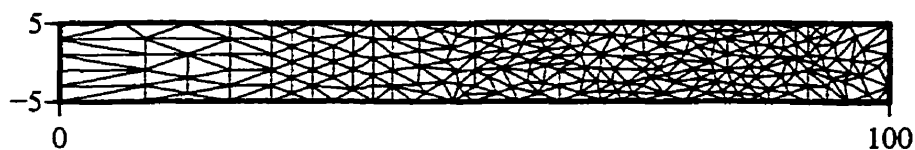
Les résultats présentés ici ont été obtenus en utilisant une longueur optimale des arêtes dans la métrique arbitraire de 0.15 mm^2 . La figure 4.2, montre le maillage initial et les résultats obtenus après les trois premières itérations du processus d'adaptation. Ces résultats ont été obtenus en prenant comme mesure d'erreur la dérivée seconde du déplacement résultant en un nœud ($\sqrt{u^2 + v^2}$).



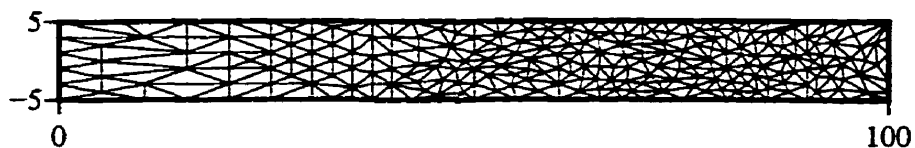
(a) Maillage initial



(b) Maillage après la première itération



(c) Maillage après la deuxième itération



(d) Maillage après la troisième itération

FIGURE 4.2: Maillages obtenus au cours du processus d'adaptation

Les résultats montrés à la figure 4.2 indiquent que les éléments tendent à s'approcher de l'encastrement. Étant donné que le moment fléchissant varie linéairement entre l'encastrement (à droite) et l'extrémité libre (à gauche) où il est nul, le processus d'adaptation concentre les éléments là où le moment fléchissant est le plus grand. Le processus d'adaptation tend donc à raffiner le maillage là où les plus importantes variations des déplacements surviennent.

Le tableau 4.1 présente les résultats obtenus à la suite du calcul de l'erreur sur chacune des arêtes pour les différents maillages obtenus. Il est important de remarquer que les résultats présentés pour une itération sont ceux du maillage en début d'itération. Par exemple, les résultats donnés pour l'itération 1 sont ceux obtenus du maillage initial, soit celui présenté à la figure 4.2(a).

TABLEAU 4.1: Calcul d'erreur sur les arêtes

		fig. 4.2(a)	fig. 4.2(b)	fig. 4.2(c)
ψ_i	moyenne	0.4762	0.8995	0.9185
	écart-type	0.0852	0.0683	0.0517
Erreur	moyenne (mm^2)	0.3063	0.0305	0.0148
	maximum (mm^2)	1.0428	0.2607	0.1829
Flèche à l'extrémité ($v_{ex} = -0.02 mm$)		-0.0090	-0.0172	-0.0180
Nombre de nœuds		66	153	248

On voit bien, à partir des résultats du tableau 4.1, que la moyenne des ψ_i tend vers l'unité. Ce fait démontre que la solution éléments finis s'approche du plus en plus de la solution exacte. Ceci implique évidemment que l'erreur tend vers zéro. Aussi, l'écart-type des ψ_i tend à diminuer, ce qui montre que le processus d'adaptation tend à uniformiser l'erreur. On voit donc que le processus d'adaptation tend non seulement à réduire l'erreur sur tout le domaine mais tend également à uniformiser l'erreur sur celui-ci.

La figure 4.3 présente l'évolution de l'erreur moyenne sur les arêtes en fonction du nombre de nœuds pour chacun des maillages obtenus au cours du processus d'adaptation. Afin de comparer les maillages adaptés à ceux obtenus en raffinant

uniformément, la courbe de l'erreur moyenne sur les arêtes en fonction du nombre de nœuds est donnée pour les deux cas. En comparant les deux courbes, on remarque que, pour un même nombre de nœuds, les maillages adaptés produisent toujours des résultats plus précis que ceux qui sont uniformément raffinés.

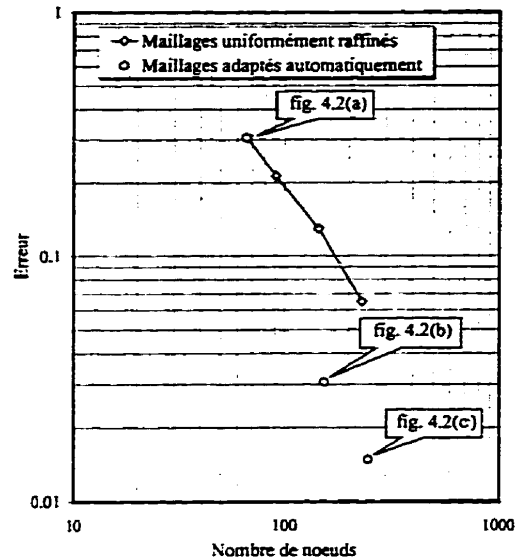


FIGURE 4.3: Erreur moyenne sur les arêtes vs nombre de nœuds

Les maillages montrés jusqu'à maintenant ont tous été obtenus au cours du même processus d'adaptation, pour lequel la longueur optimale dans la métrique est gardée constante. Nous allons maintenant étudier l'effet de varier la longueur optimale dans la métrique sur le maillage final obtenu. La figure 4.4 présente les maillages finaux obtenus pour différentes longueurs optimales dans la métrique. Aussi, le tableau 4.2 présente quelques résultats pour chacun des maillages et la figure 4.5 montre les courbes de convergence pour les cas adaptés et uniformément raffinés. Le choix de longueurs optimales dans la métrique de 0.25 mm^2 , 0.2 mm^2 et 0.15 mm^2 est arbitraire.

On a démontré, dans le chapitre 3, que la longueur dans la métrique est directement reliée à l'erreur d'interpolation. On remarque donc très clairement que pour un niveau d'erreur plus élevé (longueur optimale dans la métrique plus élevée), le maillage final obtenu est plus grossier. On remarque aussi que la courbe de convergence pour les maillages uniformes non adaptés est parallèle à celle obtenue pour les

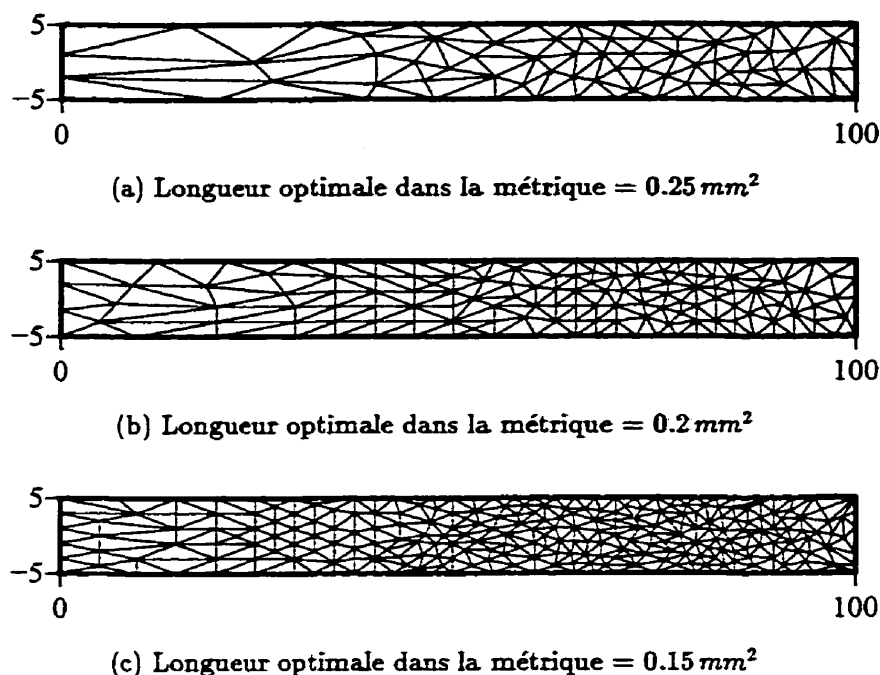


FIGURE 4.4: Maillages obtenus pour différentes longueurs optimales dans la métrique

TABLEAU 4.2: Résultats vs longueur optimale dans la métrique

Longueur optimale dans la métrique (mm^2)	0.25	0.2	0.15
Moyenne de l'erreur sur les arêtes (mm^2)	0.0742	0.0341	0.0148
Nombre de nœuds	66	153	248
Déflexion (mm)	-0.0144	-0.0166	-0.0180

maillages adaptés puisque la pente de la courbe de convergence est conditionné par l'élément utilisé et non par le maillage.

4.2 Cas 2 : Poutre fléchie en porte-à-faux

La plaque à l'étude (voir figure 4.6) a une épaisseur t de 1 mm et est chargée par une pression uniforme appliquée sur le domaine entier. L'intensité de la pression appliquée q est de 0.001 N/mm^2 . La plaque est constituée d'un matériau isotrope dont le module d'Young E est de 200000 MPa et le coefficient de Poisson ν est 0.3 . La longueur de la poutre L est 100 mm et sa hauteur h est 10 mm .

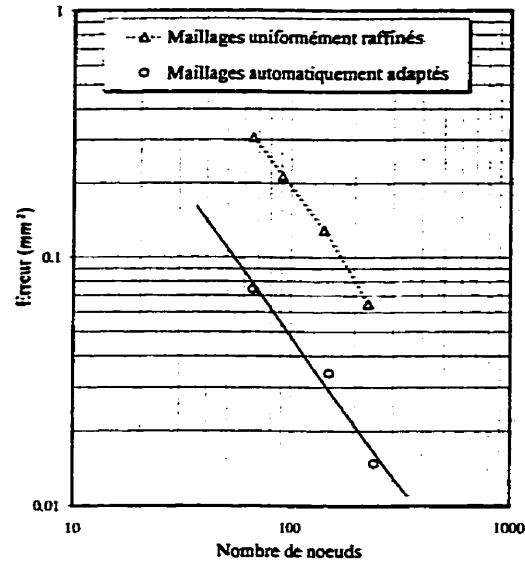


FIGURE 4.5: Courbe de convergence pour le cas 1

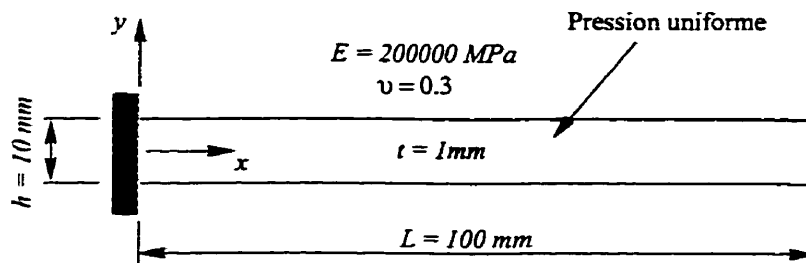


FIGURE 4.6: Description du modèle

Le déplacement transverse w pour une poutre en porte-à-faux soumise à une pression uniforme est donné par :

$$w(x, y) = \frac{1}{24} \frac{qx^2(x^2 + 6L^2 - 4Lx)}{EI} \quad (4.4)$$

où I représente l'inertie autour de l'axe y . Il est important de remarquer que la solution est indépendante de y . Il n'y aura donc pas de variation de la solution suivant cette direction. À noter que cette solution n'est valide que pour une poutre mince qui respecte l'hypothèse de Bernoulli. À titre indicatif, la flèche au bout du porte-à-faux calculée à l'aide de l'équation 4.4 est de 0.75 mm .

4.2.1 Résultats de l'adaptation de maillage

L'adaptation du maillage du problème décrit à la section précédente a causé quelques difficultés. Afin de bien cerner ces difficultés, le processus d'adaptation sera fait en deux temps. Premièrement, le processus d'adaptation sera effectué sans restreindre la longueur des arêtes. C'est-à-dire que le déraffinement des arêtes sera libre de s'effectuer sans restriction quant à la longueur maximale d'une arête. Dans un deuxième temps, nous limiterons la longueur des arêtes à différents niveaux afin de bien analyser l'influence de cette limite sur l'adaptation du maillage.

4.2.1.1 Résultats sans restriction sur la longueur des arêtes

Les résultats présentés ici ont été obtenus en utilisant une longueur optimale des arêtes dans la métrique de 0.05 mm^2 . Les figures 4.7(a) à 4.7(e) montrent les résultats obtenus pour les quatre premières itérations. Le tableau 4.3 présente les résultats du calcul d'erreur pour chacun des maillages obtenus. Ces résultats ont été obtenus en prenant comme mesure d'erreur la dérivée seconde du déplacement résultant (w).

TABLEAU 4.3: Calcul d'erreur sur les arêtes (sans restriction)

		itération 1	itération 2	itération 3	itération 4
ψ_i	moyenne	1.0068	1.0295	1.0141	1.0095
	écart-type	0.1072	0.2402	0.1204	0.1129
Erreur	moyenne (mm^2)	0.0472	0.0168	0.0149	0.0138
	maximum (mm^2)	0.1275	0.1838	0.1136	0.0930
Flèche ($w_{ez} = -0.75 \text{ mm}$)		-0.7370	-0.7426	-0.7442	-0.7469
Nombre de nœuds		66	46	40	40

Sur les figures 4.7(a) à 4.7(e), l'encastrement est à gauche. On voit très bien sur ces figures que les éléments tendent à s'approcher de l'encastrement. Étant donné que pour un tel problème le moment fléchissant varie de manière quadratique entre l'encastrement où il est maximum et l'extrémité libre où il est nulle, on voit très

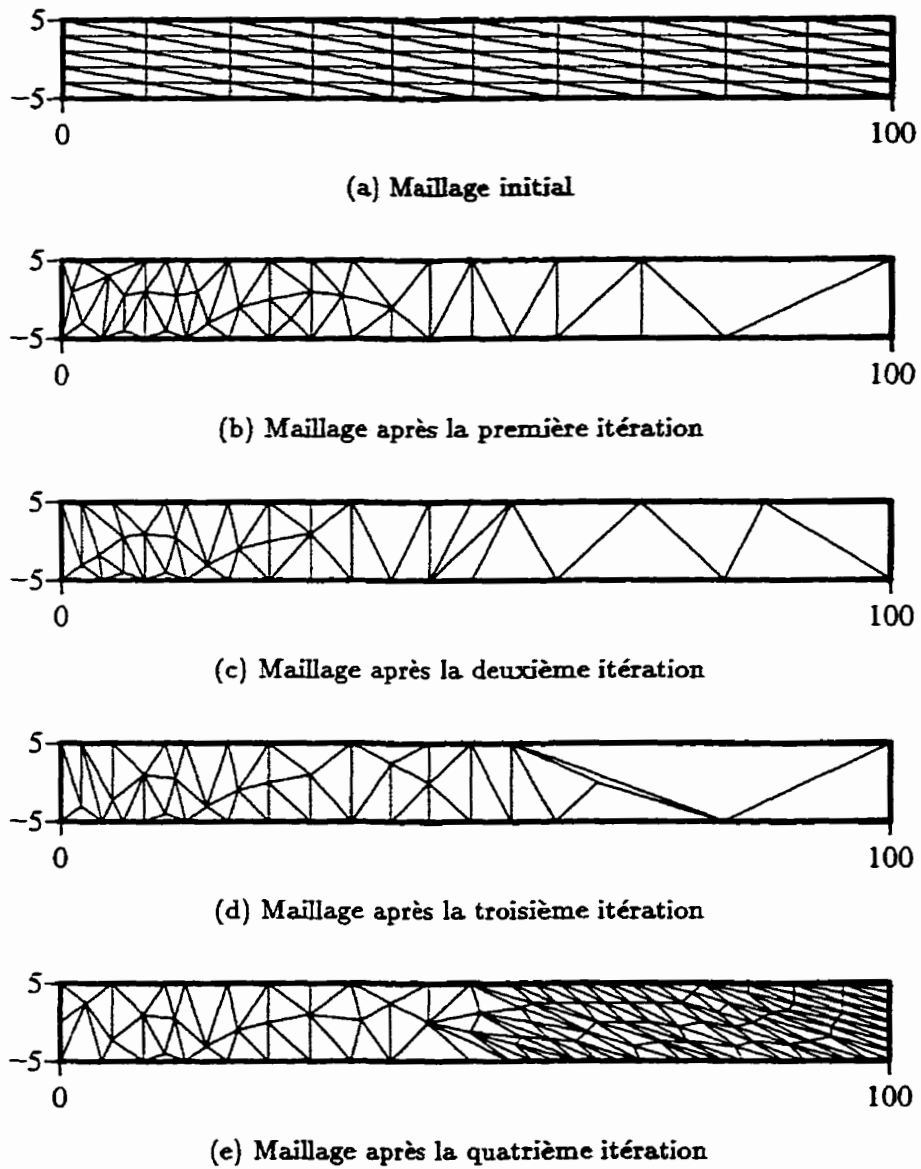


FIGURE 4.7: Maillages obtenus au cours du processus d'adaptation (sans restriction)

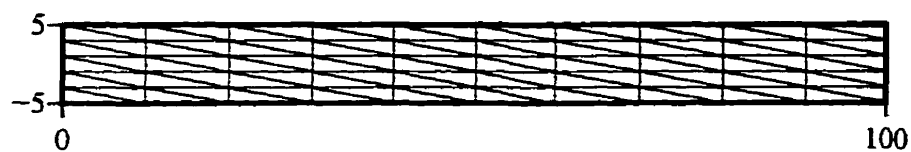
bien que le processus d'adaptation concentre les éléments dans les zones de plus fort moment fléchissant. On voit aussi que, puisque pour ce cas de chargement la variation de la solution suivant la direction y est nulle (dérivée seconde nulle), la dimension des éléments suivant cette direction tend à être maximale, c'est-à-dire égale à la largeur du porte-à-faux.

Par contre, on remarque que le maillage obtenu après l'itération quatre est inacceptable. Cette situation est causée par le fait que sur les frontières du domaine, la dérivée seconde du critère d'adaptation ne peut être calculée (voir section 3.1.2). La stratégie alors utilisée par "Libom" est que, pour un nœud situé sur une frontière, la dérivée seconde est prise égale à celle du nœud le plus près situé à l'intérieur du domaine. On voit donc que pour le problème analysé, tous les nœuds situés dans la zone de l'extrémité libre du porte-à-faux sont situés sur les frontières du domaine et que pour ces nœuds la dérivée seconde utilisée n'a pas la bonne valeur. Ceci mène à une très mauvaise adaptation.

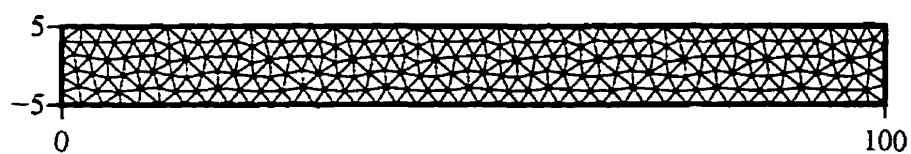
Afin de contourner cette difficulté, "Libom" permet de restreindre la dimension des arêtes pour tous les éléments. La section suivante présente les résultats obtenus pour différentes limites choisies.

4.2.1.2 Résultats avec restriction sur la dimension des arêtes

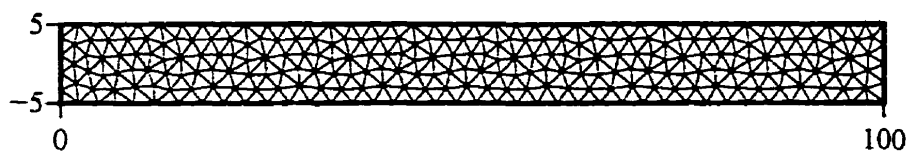
Les résultats présentés ici ont encore été obtenus en utilisant une longueur optimale des arêtes dans la métrique de 0.05 mm^2 . La figure 4.8 montre les maillages obtenus pour une limite sur la dimension des arêtes égale à 3.33 mm . La figure 4.9 montre les maillages obtenus pour une limite sur la dimension des arêtes égale à 5.0 mm . Finalement, la figure 4.10 montre les maillages obtenus pour une limite sur la dimension des arêtes égale à 10.0 mm . Ces résultats ont toujours été obtenus en prenant comme critère d'adaptation le déplacement w . Afin de pouvoir bien comparer les différents maillages, les tableaux 4.4, 4.5 et 4.6 présentent les résultats sur les calculs d'erreur pour les différents maillages.



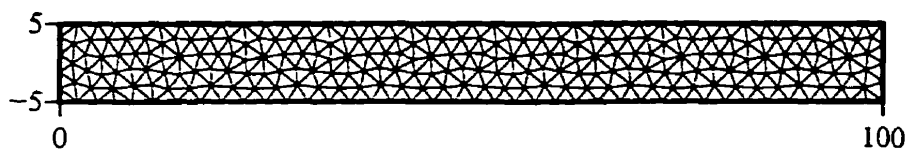
(a) Maillage initial



(b) Maillage après la première itération

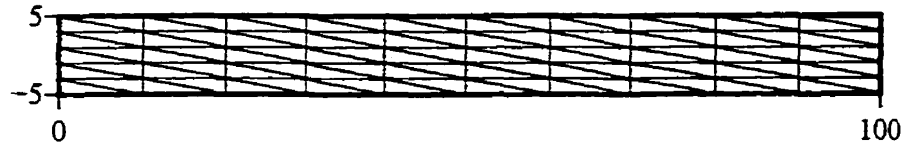


(c) Maillage après la deuxième itération

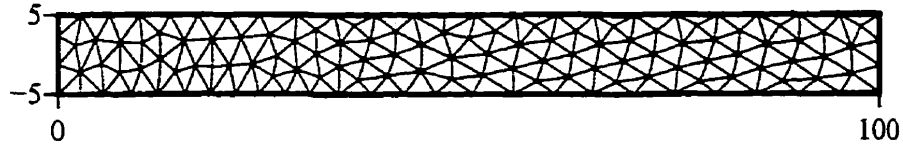


(d) Maillage après la troisième itération

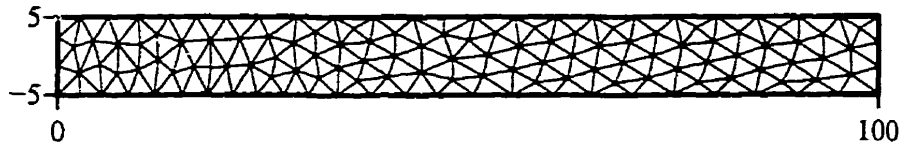
FIGURE 4.8: Maillages obtenus au cours du processus d'adaptation (limite = 3.33 mm)



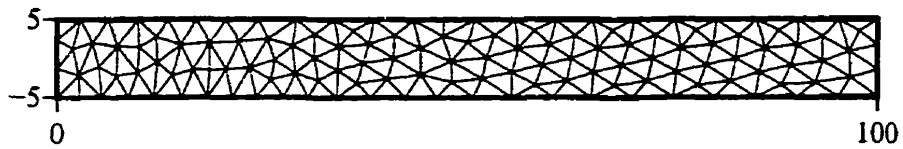
(a) Maillage initial



(b) Maillage après la première itération



(c) Maillage après la deuxième itération



(d) Maillage après la troisième itération

FIGURE 4.9: Maillages obtenus au cours du processus d'adaptation (limite = 5.0 mm)

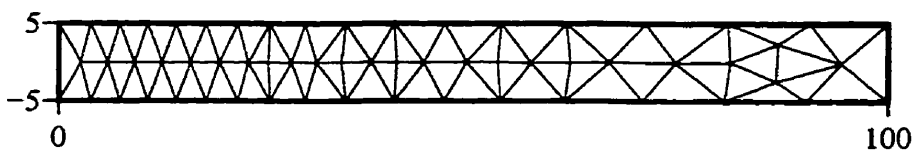
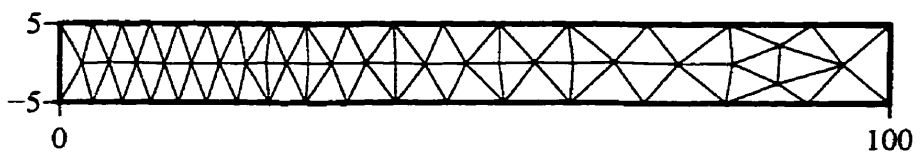
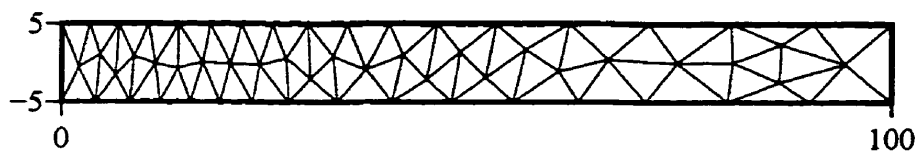
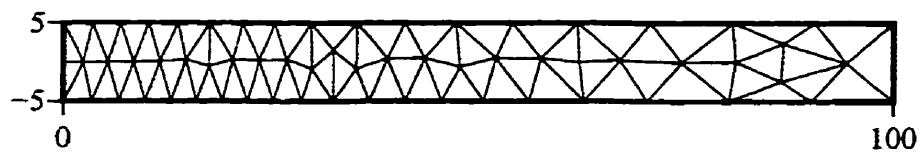
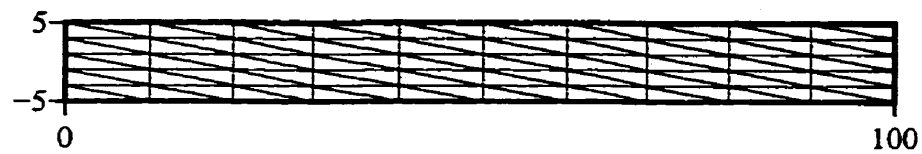


FIGURE 4.10: Maillages obtenus au cours du processus d'adaptation (limite = 10.0 mm)

TABLEAU 4.4: Calcul d'erreur sur les arêtes (limite = 3.33 mm)

		itération 1	itération 2	itération 3
ψ_i	moyenne	1.0068	0.9839	0.9829
	écart-type	0.1072	0.0567	0.0545
Erreur	moyenne (mm^2)	0.0471	0.0105	0.0106
	maximum (mm^2)	0.1275	0.0278	0.0278
Flèche à l'extrémité ($w_{ex} = -0.75 mm$)		-0.7370	-0.7406	-0.7406
Nombre de nœuds		66	275	274

TABLEAU 4.5: Calcul d'erreur sur les arêtes (limite = 5.0 mm)

		itération 1	itération 2	itération 3
ψ_i	moyenne	1.0068	0.9865	0.9866
	écart-type	0.1072	0.0653	0.0661
Erreur	moyenne (mm^2)	0.0471	0.0160	0.0160
	maximum (mm^2)	0.1275	0.0475	0.0475
Flèche à l'extrémité ($w_{ex} = -0.75 mm$)		-0.7370	-0.7402	-0.7403
Nombre de nœuds		66	137	137

4.2.2 Discussion

Le graphique 4.11 présente la moyenne de l'erreur sur les arêtes en fonction du nombre de nœuds pour chacune des analyses qui ont été faites. On peut bien voir à partir de ce graphique que les solutions obtenues des maillages ayant subi un processus d'adaptation sont supérieures à la solution obtenue des maillages structurés uniformes.

À partir d'un même maillage de départ et pour des niveaux de restriction différents, on voit bien que le processus d'adaptation converge vers des maillages différents. On remarque que l'imposition de restrictions sur la longueur des arêtes réduit l'efficacité du processus d'adaptation. À partir de cette constatation, il apparaît important de pouvoir mieux mesurer les dérivées secondes sur les frontières afin d'avoir une meilleure mesure de l'erreur sur les frontières et ainsi éviter d'imposer des limites inutiles. Dans des travaux futurs, il serait donc primordial de pouvoir

TABLEAU 4.6: Calcul d'erreur sur les arêtes (limite = 10.0 mm)

		itération 1	itération 2	itération 3	itération 4
ψ_i	moyenne	1.0068	0.9930	1.0050	1.0003
	écart-type	0.1072	0.0999	0.1456	0.1180
Erreur	moyenne (mm^2)	0.0471	0.0222	0.0198	0.0199
	maximum (mm^2)	0.1275	0.0948	0.0835	0.0823
Flèche ($w_{ez} = -0.75 mm$)		-0.7370	-0.7405	-0.7417	-0.7418
Nombre de nœuds		66	64	60	60

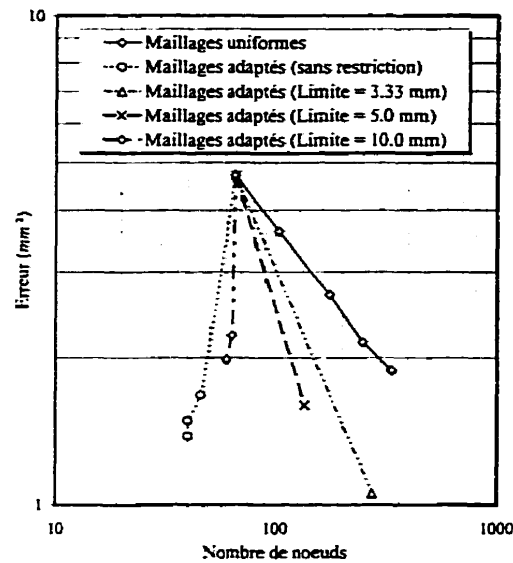


FIGURE 4.11: Erreur vs nombre de nœuds

mesurer l'erreur de façon adéquate sur les frontières du domaine de façon à éviter les problèmes rencontrés ici.

Malgré ceci, nous pouvons conclure que la librairie d'adaptation de maillage "Libom" nous donne d'excellents résultats. En effet, en comparaison avec les maillages structurés uniformes, les maillages adaptés obtenus sont de beaucoup supérieurs.

La prochaine section présente un problème très similaire à celui étudié ici afin de voir si les difficultés rencontrées pour une plaque en porte-à-faux sont également présentes pour une plaque soumise à d'autres conditions aux limites.

4.3 Cas 3 : Poutre fléchie simplement supportée

La figure 4.12 présente la géométrie et les propriétés du modèle analysé. La plaque à l'étude a une épaisseur t de 1 mm et est sollicitée par une pression q uniformément répartie sur le domaine entier. L'intensité de la charge q est de 10.0 N/mm^2 . La plaque est constituée d'un matériau isotrope dont le module d'Young E est de 200000 MPa et le coefficient de Poisson ν est 0.3 . La poutre a une portée L de 100 mm et une largeur h de 10 mm .

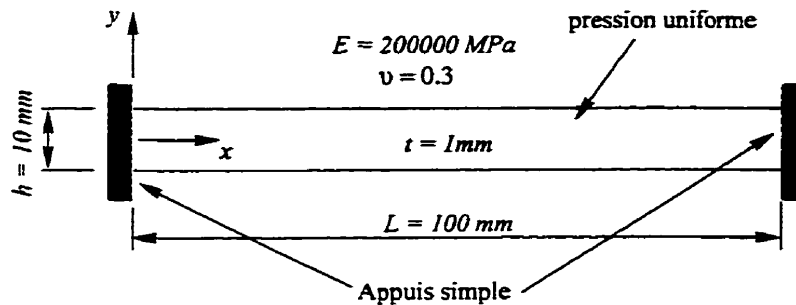


FIGURE 4.12: Description du modèle

Le déplacement transverse w pour une telle poutre est donné par :

$$w(x, y) = \frac{1}{24} \frac{qx(x^3 + L^3 - 2Lx^2)}{EI} \quad (4.5)$$

où I représente l'inertie autour de l'axe y . Il est important de remarquer que la solution est indépendante de y . Il n'y aura donc pas de variation de la solution suivant cette direction. À titre indicatif, la flèche au centre de la poutre calculée à l'aide de l'équation 4.5 est de 0.7813 mm . À noter que ce résultat a été obtenu en utilisant l'hypothèse de Bernoulli.

4.3.1 Résultats de l'adaptation de maillage

Tout comme le problème du porte-à-faux de la section 4.2, l'adaptation du maillage pour ce problème a causé quelques difficultés. Afin de bien cerner ces difficultés, le processus d'adaptation sera fait en deux temps. Premièrement, le processus d'adaptation sera effectué sans restreindre la longueur des arêtes. C'est-à-dire

que le déraffinement des arêtes sera libre de s'effectuer sans restriction quant à la longueur maximale d'une arête. Dans un deuxième temps, nous limiterons la longueur des arêtes à différents niveaux afin de bien analyser l'influence de cette limite sur l'adaptation du maillage pour le présent cas.

4.3.1.1 Résultats sans restriction sur la longueur des arêtes

Les résultats présentés ici ont été obtenus en utilisant une longueur optimale des arêtes dans la métrique de 0.075 mm^2 . Les figures 4.13(a) à 4.13(e) montrent le maillage initial et les résultats obtenus pour les quatre premières itérations. Le tableau 4.7 présente les résultats du calcul d'erreur pour chacun des maillages obtenus. Ces résultats ont été obtenus en prenant comme mesure d'erreur la dérivée seconde du déplacement résultant (w).

TABLEAU 4.7: Calcul d'erreur sur les arêtes (sans restriction)

		itération 1	itération 2	itération 3	itération 4
ψ_i	moyenne	0.9786	0.9980	0.9983	0.9985
	écart-type	0.0057	0.0014	0.0014	0.0013
Erreur	moyenne (mm^2)	0.0870	0.0046	0.0039	0.0037
	maximum (mm^2)	0.1930	0.0321	0.0230	0.0309
Flèche ($w_{ez} = -0.7813 \text{ mm}$)		-0.7708	-0.7820	-0.7823	-0.7821
Nombre de nœuds		66	93	91	89

On voit très bien sur ces figures que les éléments tendent à se regrouper au centre de la plaque. Étant donné que pour un tel problème le moment fléchissant varie quadratiquement et atteint une valeur maximum au centre de la poutre, on voit très bien que le processus d'adaptation concentre les éléments dans les zones de plus fort moment fléchissant. On voit aussi que, puisque la variation de la solution suivant la direction y est nulle (dérivée seconde nulle), la dimension des éléments suivant cette direction tend à être maximale, c'est-à-dire égale à la largeur h de la poutre.

Malgré le fait que les maillages obtenus sans restriction sur la dimension des

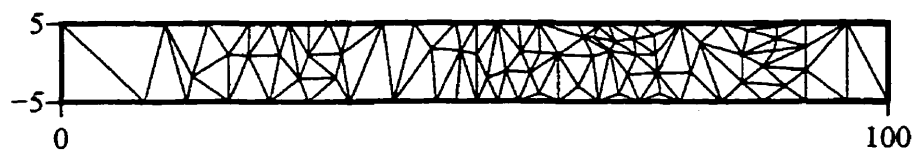
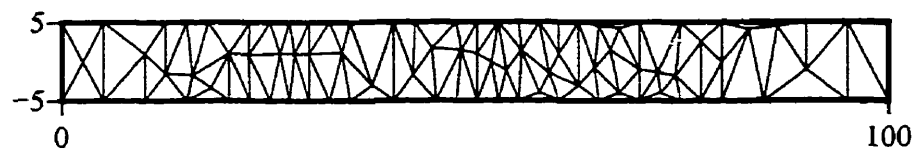
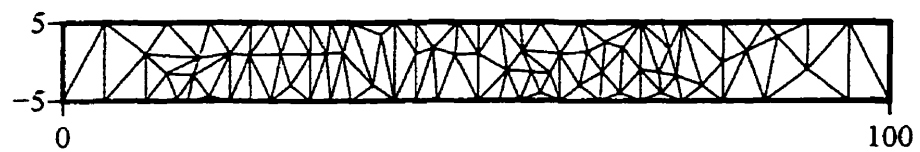
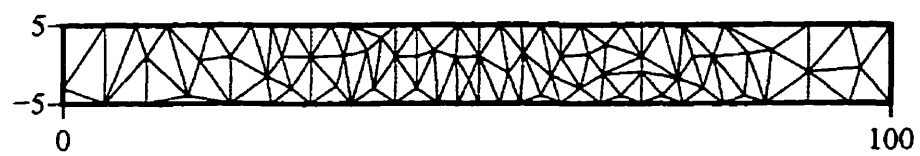
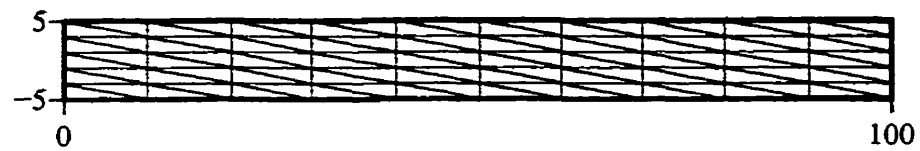


FIGURE 4.13: Maillages obtenus au cours du processus d'adaptation (sans restriction)

arêtes ne soit pas aussi mauvais que ceux obtenus dans le cas du porte-à-faux examiné à la section 4.2, les mêmes difficultés pourraient être rencontrées puisque les nœuds des éléments tendent à se retrouver sur les frontières. La section suivante présente donc les résultats obtenus en contraignant la longueur des arêtes à une valeur maximale afin de contourner ces difficultés.

4.3.1.2 Résultats avec restriction sur la dimension des arêtes

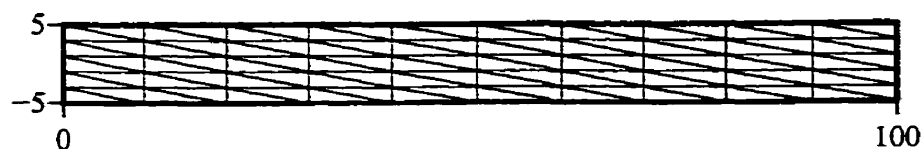
Les résultats présentés ici ont encore été obtenus en utilisant une longueur optimale des arêtes dans la métrique de 0.075 mm^2 . La figure 4.14 montrent les maillages obtenus pour une limite sur la dimension des arêtes égale à 5.0 mm et la figure 4.15 montrent les maillages obtenus pour une limite sur la dimension des arêtes égale à 10.0 mm . Ces résultats ont toujours été obtenus en prenant comme mesure d'erreur la dérivée seconde du déplacement résultant (w). Afin de pouvoir bien comparer les différents maillages, les tableaux 4.8 et 4.9 présentent les résultats sur les calculs d'erreur pour les différents maillages.

TABLEAU 4.8: Calcul d'erreur sur les arêtes (limite = 5.0 mm)

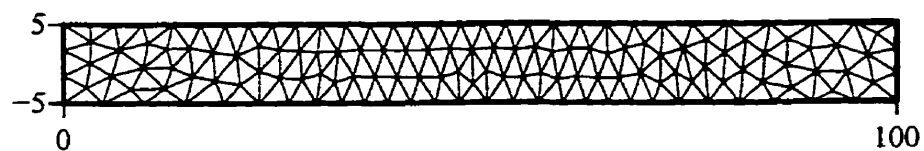
		itération 1	itération 2	itération 3
ψ_i	moyenne	0.9786	0.9985	0.9985
	écart-type	0.0057	0.0011	0.0011
Erreur	moyenne (mm^2)	0.0869	0.0023	0.0023
	maximum (mm^2)	0.1930	0.0063	0.0063
Flèche à l'extrémité ($w_{ez} = -0.7813 \text{ mm}$)		-0.7708	-0.7821	-0.7823
Nombre de nœuds		66	146	147

4.3.2 Discussion

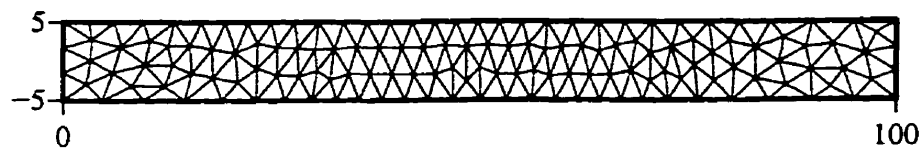
Le graphique 4.16 présente la moyenne de l'erreur sur les arêtes en fonction du nombre de nœuds pour chacune des analyses qui ont été faites. Ces résultats permettent de conclure que les solutions obtenues des maillages ayant subi un processus d'adaptation sont plus précis que ceux obtenus des maillages structurés uniformes.



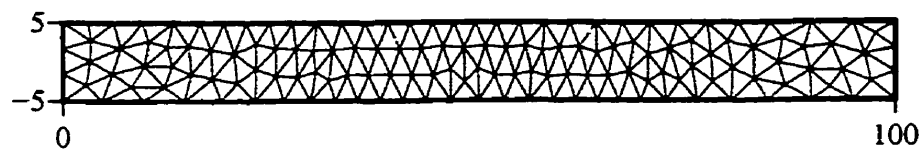
(a) Maillage initial



(b) Maillage après la première itération

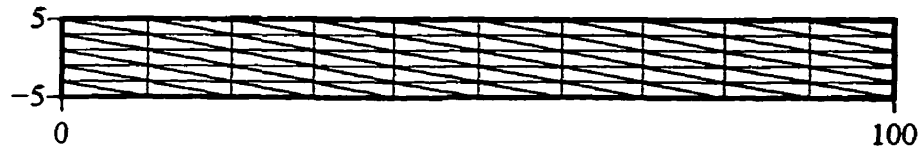


(c) Maillage après la deuxième itération

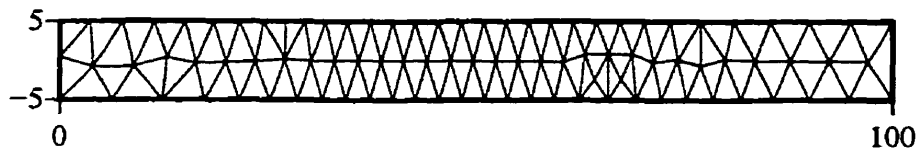


(d) Maillage après la troisième itération

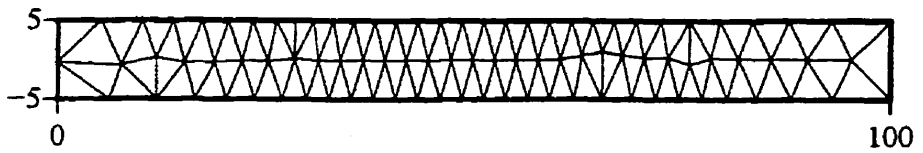
FIGURE 4.14: Maillages obtenus au cours du processus d'adaptation (limite = 5.0 mm)



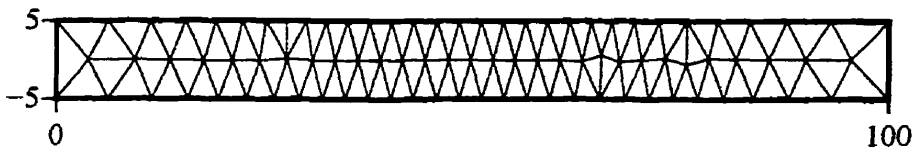
(a) Maillage initial



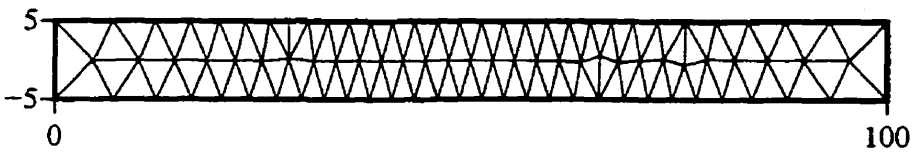
(b) Maillage après la première itération



(c) Maillage après la deuxième itération



(d) Maillage après la troisième itération



(e) Maillage après la quatrième itération

FIGURE 4.15: Maillages obtenus au cours du processus d'adaptation (limite = 10.0 mm)

TABLEAU 4.9: Calcul d'erreur sur les arêtes (limite = 10.0 mm)

		itération 1	itération 2	itération 3	itération 4
ψ_i	moyenne	0.9786	0.9989	0.9989	0.9990
	écart-type	0.0057	0.0012	0.0013	0.0013
Erreur	moyenne (mm^2)	0.0869	0.0025	0.0026	0.0023
	maximum (mm^2)	0.1930	0.0067	0.0062	0.0061
Flèche ($w_{ex} = -0.7813 mm$)		-0.7708	-0.7820	-0.7818	-0.7819
Nombre de nœuds		66	94	93	91

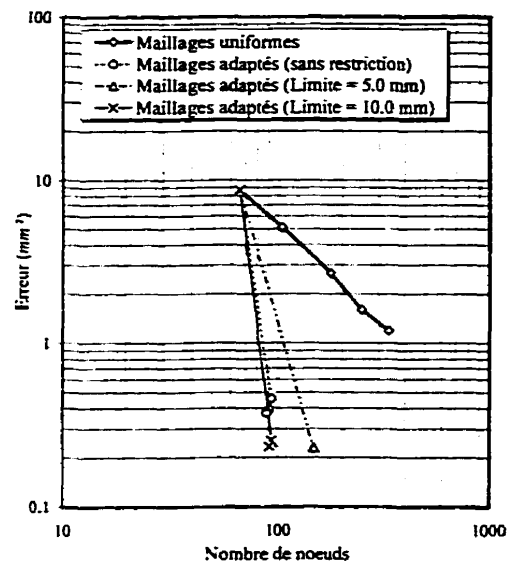


FIGURE 4.16: Erreur vs nombre de nœuds

Étant donné que ce problème comporte les mêmes caractéristiques que celles rencontrées lors de la résolution du cas 2 de la section 4.2, les mêmes commentaires peuvent être faits. Le lecteur est donc invité à consulter la section 4.2.2 pour plus de détails.

4.4 Cas 4 : Plaque carrée orthotrope simplement supportée

4.4.1 Géométrie et propriétés du modèle

Dans cette section, nous examinerons le cas d'une plaque carrée ayant des côtés de longueur "a" de 1000 mm, soumise à une pression q uniforme de 64.5 Pa. La figure 4.17 présente la géométrie et les propriétés du modèle analysé. La plaque à l'étude a une épaisseur totale de 6.8 mm et est constituée d'un stratifié symétrique comportant cinq couches disposées selon la figure 4.17(b). Les couches 1, 3 et 5 sont des couches à renfort mat ayant les caractéristiques suivantes :

$$E_L = E_T = 7.72 \text{ GPa} \quad \nu_{LT} = 0.33 \quad G_{LT} = 2.91 \text{ Gpa} \quad (4.6)$$

La couche 3 est une couche double. Les couches 2 et 4 sont des couches doubles à renfort tissu dont les caractéristiques sont les suivantes :

$$E_L = E_T = 13.8 \text{ GPa} \quad \nu_{LT} = 0.12 \quad G_{LT} = 1.87 \text{ Gpa} \quad (4.7)$$

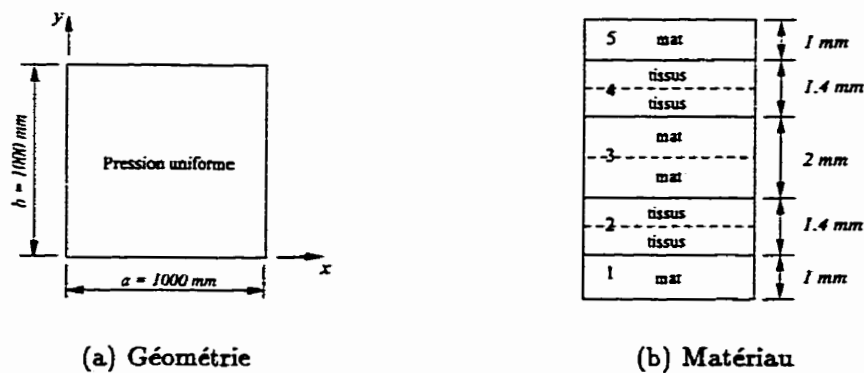


FIGURE 4.17: Description du modèle

Le déplacement transverse w d'une plaque orthotrope et simplement supportée sur tout le contour soumise à une charge uniformément répartie est donné par la relation suivante [2] :

$$w(x, y) = \frac{16qa^4}{\pi^6} \sum_{m=1}^{\infty} \sum_{n=1}^{\infty} c_{mn} \sin(2m-1) \frac{\pi x}{a} \sin(2n-1) \frac{\pi y}{a} \quad (4.8)$$

avec

$$c_{mn} = \frac{1}{(2m-1)(2n-1)D_{2m-1, 2n-1}} \quad (4.9)$$

Les coefficients D_{ij} de l'équation 4.9 sont donnés par :

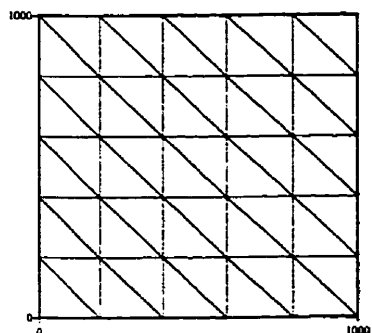
$$D_{mn} = H_{f_{11}} m^4 + 2(H_{f_{12}} + 2H_{f_{33}})(mn)^2 + H_{f_{22}} n^4 \quad (4.10)$$

où les coefficients $H_{f_{ii}}$ sont les termes de la matrice H_f (voir l'annexe A).

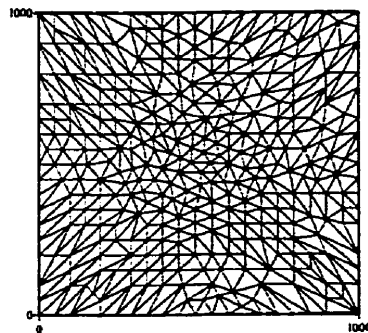
4.4.2 Résultats de l'adaptation de maillage

Les résultats présentés ici ont été obtenus en utilisant une longueur optimale des arêtes dans la métrique (LoptMet) de 0.15 mm^2 . La figure 4.18 montre le maillage initial et les résultats obtenus pour les quatres premières itérations. Ces résultats ont été obtenus en prenant comme mesure d'erreur la dérivée seconde du déplacement transverse w et ceux-ci ont été obtenus sans aucune restriction suivant la longueur des arêtes ou l'aspect géométrique des éléments.

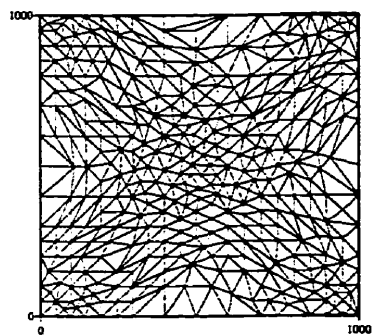
On voit très bien sur les maillages de la figure 4.18 que les éléments tendent à se regrouper vers le centre de la plaque ainsi que dans les coins. Ceci n'est pas surprenant lorsque l'on regarde l'allure des dérivées secondes exactes montrées à la figure 4.19. En effet, on remarque que les dérivées $\frac{\partial^2 w}{\partial x^2}$ et $\frac{\partial^2 w}{\partial y^2}$ sont identiques et que celles-ci sont maximales au centre de la plaque et nulles sur les frontières. Ces dérivées sont donc la cause du regroupement des éléments dans la région du centre de la plaque. Aussi, on remarque que les dérivées $\frac{\partial^2 w}{\partial x \partial y}$ sont nulles au centre de la plaque et maximales en valeur absolue dans les coins. C'est donc la dérivée $\frac{\partial^2 w}{\partial x \partial y}$ qui est responsable du regroupement des éléments dans les coins. On remarque également que les valeurs numériques maximales pour les dérivées $\frac{\partial^2 w}{\partial x^2}$, $\frac{\partial^2 w}{\partial y^2}$ et $\frac{\partial^2 w}{\partial x \partial y}$ sont identiques ; la dimension des éléments dans la zone centrale et dans les coins sont donc les mêmes. De plus, on obtient un maillage isotrope dans la zone centrale puisque $\frac{\partial^2 w}{\partial x^2} = \frac{\partial^2 w}{\partial y^2}$ et que $\frac{\partial^2 w}{\partial x \partial y}$ est nulle.



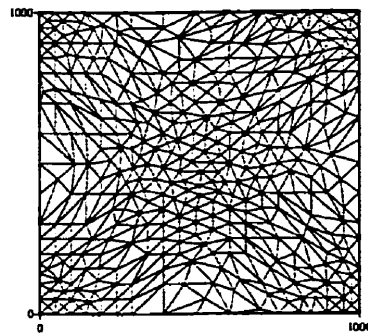
(a) Maillage initial



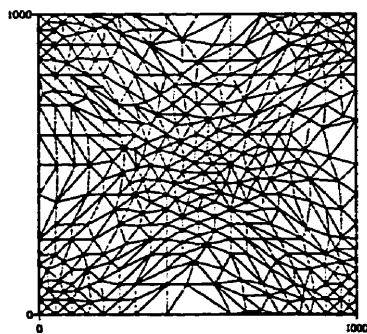
(b) Maillage après la première itération



(c) Maillage après la deuxième itération



(d) Maillage après la troisième itération



(e) Maillage après la quatrième itération

FIGURE 4.18: Maillages obtenus au cours du processus d'adaptation

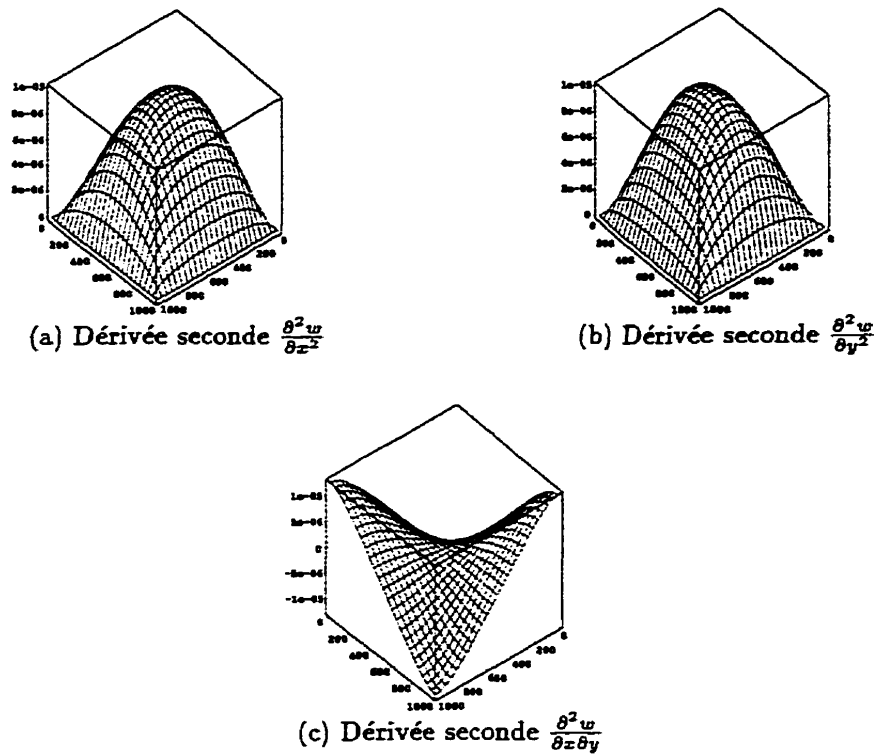


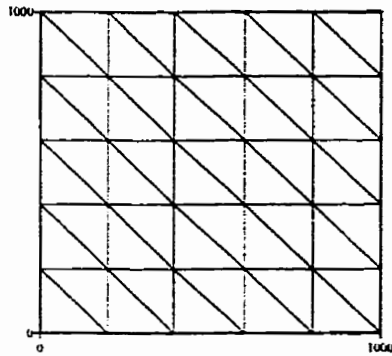
FIGURE 4.19: Graphiques des dérivées secondes

Le tableau 4.10 présente les résultats obtenus à la suite du calcul de l'erreur sur chacune des arêtes pour les différents maillages obtenus.

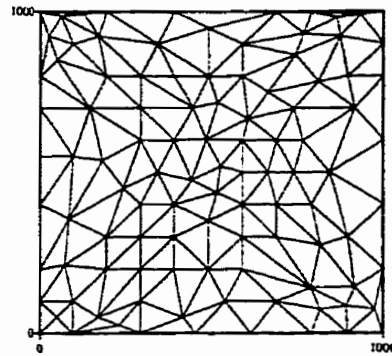
Tous les résultats présentés jusqu'à maintenant ont été obtenus avec la même longueur optimale dans la métrique, soit une longueur de 0.15 mm^2 . Les résultats qui suivent ont été obtenus en faisant varier cette longueur afin de vérifier comment le nombre de nœuds obtenus pour les maillages adaptés varie en fonction du niveau choisi pour la longueur optimale dans la métrique.

Les figures 4.20(b), 4.20(c) et 4.20(d) présentent les résultats obtenus pour des longueurs optimales dans la métrique de 0.3 mm^2 , 0.23 mm^2 et 0.15 mm^2 respectivement. Le tableau 4.11 présente quelques résultats relatifs à ces maillages. Ces derniers sont les maillages obtenus à la suite d'un processus complet d'adaptation.

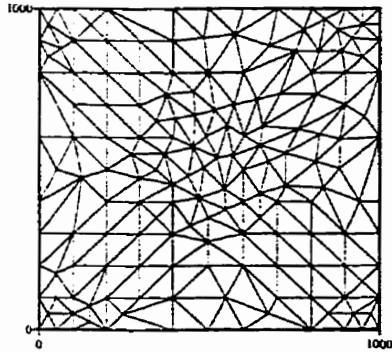
On voit sur la figure 4.20 que pour des choix différents de la longueur optimale dans la métrique, on obtient toujours le même patron d'éléments. Aussi, on remarque



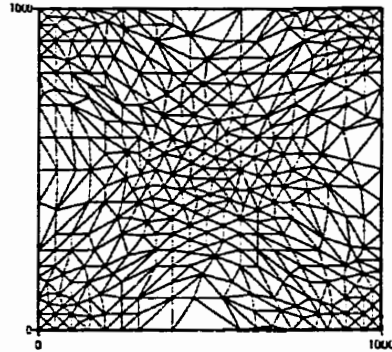
(a) Maillage uniforme



(b) Maillage adapté ($L_{optMet} = 0.3 \text{ mm}^2$)



(c) Maillage adapté ($L_{optMet} = 0.23 \text{ mm}^2$)



(d) Maillage adapté ($L_{optMet} = 0.15 \text{ mm}^2$)

FIGURE 4.20: Maillages obtenus pour différente longueur optimale dans la métrique

TABLEAU 4.10: Calcul d'erreur sur les arêtes (LoptMet = 0.15 mm²)

		itération 1	itération 2	itération 3	itération 4
ψ_i	moyenne	0.8944	1.0122	0.9992	0.9967
	écart-type	0.5687	0.2167	0.2105	0.1882
Erreur	moyenne (mm ²)	24.8481	0.7310	0.4674	0.3710
	maximum (mm ²)	129.26	23.70	34.60	34.45
Flèche au centre		-0.9855	-1.1160	-1.1134	-1.1125
Nombre de nœuds		36	356	381	398

TABLEAU 4.11: Résultats vs Longueur optimale dans la métrique

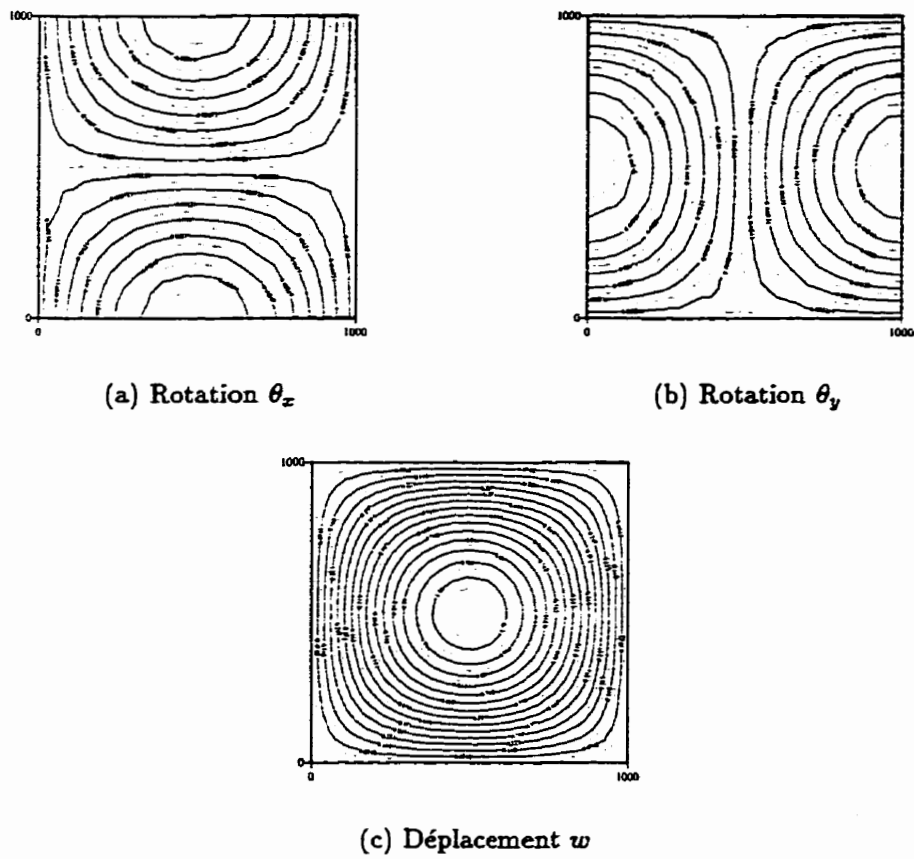
Longueur optimale dans la métrique (mm ²)	0.30	0.23	0.15
Moyenne de l'erreur sur les arêtes (mm ²)	2.2049	1.9937	0.3710
Nombre de nœuds	107	169	398
déflexion (mm)	1.1363	1.1155	1.1092

que plus la longueur optimale dans la métrique est petite, plus la densité du motif est élevée. Ainsi, pour un passage de la longueur optimale dans la métrique de 0.3 mm² à 0.15 mm², on remarque une forte différence dans la densité des éléments. Le nombre de nœuds ou encore la densité du maillage est donc très sensible par rapport au paramètre LoptMet.

4.4.3 Discussion

Afin de bien analyser les résultats obtenus, cette section présente quelques tableaux et graphiques permettant de mieux évaluer la qualité des maillages. Le graphique 4.21 montre les isolignes des composantes θ_x et θ_y de la rotation et du déplacement transverse w pour le maillage de la figure 4.18(e). Ces graphiques démontrent que les maillages adaptés permettent d'obtenir des isolignes très lisses. Le contraire aurait été d'obtenir des isolignes montrant des oscillations se qui aurait eu comme effet de diminuer la qualité des contraintes calculées puisque les oscillations auraient amené des contraintes parasites.

Le graphique 4.22 présente l'évolution de l'erreur sur les arêtes en fonction du

FIGURE 4.21: Isocontours pour les composantes θ_x , θ_y et w du déplacement

nombre de nœuds au cours d'un même processus d'adaptation. Il s'agit des résultats obtenus pour une longueur optimale dans la métrique de 0.15 mm^2 . Ce graphique démontre que le processus d'adaptation permet d'obtenir des solutions plus précises que celles obtenues des maillages structurés uniformément raffinés.

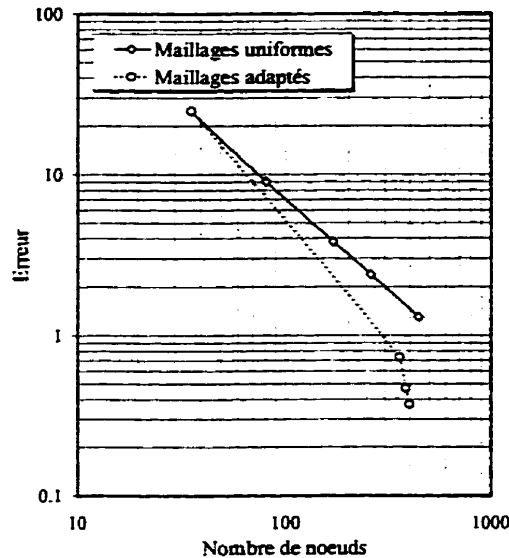


FIGURE 4.22: Erreur vs nombre de nœuds

Aussi, le graphique 4.23 présente les courbes de convergence pour les maillages uniformes et les maillages adaptés convergés. Cette figure révèle que la pente obtenue dans le cas des maillages uniformes est la même que celle obtenue dans le cas des maillages adaptés convergés. Le taux de convergence demeure donc le même puisque les éléments utilisés sont les mêmes. On remarque aussi que, pour un nombre donné de nœuds, l'erreur est inférieure pour les maillages adaptés que pour les maillages uniformes.

Tous les calculs d'erreur décrits jusqu'à maintenant ont été faits en comparant la solution obtenue par éléments finis à celle théorique. D'une manière intuitive, on pourrait aussi choisir d'estimer l'erreur en calculant, pour chaque élément, l'écart maximale entre une composante de σ obtenue pour un élément et la même composante obtenue dans chacun des éléments voisins ($\Delta\sigma_{\max}$). Cette manière d'estimer l'erreur est d'ailleurs suggérée par Bathe [3]. Le tableau 4.12 nous montre les statistiques relatives au calcul de l'écart entre deux éléments voisins de la contrainte σ_z calculée sur la surface supérieure de la plaque ($z = 0.5 \text{ mm}$). Les contraintes sont

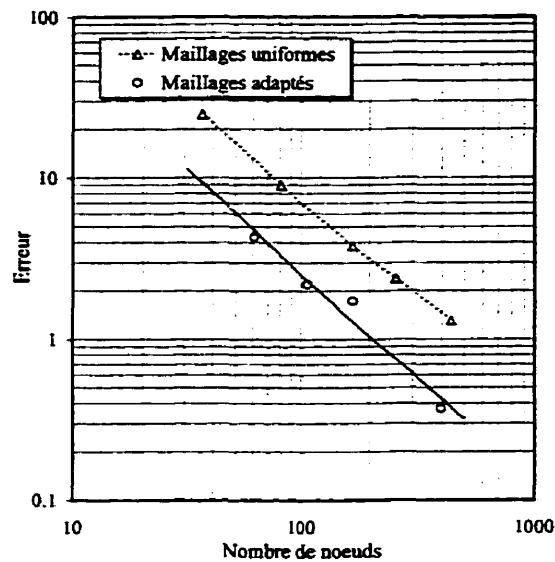


FIGURE 4.23: Précision des maillages uniformes et des maillages adaptés

calculées au centre de chacun des éléments.

 TABLEAU 4.12: Statistiques sur le calcul de la contrainte σ_x

	maillage figure 4.18(a)	maillage figure 4.18(d)
Moyenne des $\Delta\sigma_x$ (MPa)	0.0141	0.0048
Maximum des $\Delta\sigma_x$ (MPa)	0.0421	0.0267
Écart-type des $\Delta\sigma_x$ (MPa)	0.0101	0.0041

Il est donc extrêmement intéressant de remarquer que le processus permet une meilleure prédiction des contraintes en diminuant de façon très significative l'écart qui peut exister entre les contraintes pour deux éléments voisins. À remarquer que les mêmes conclusions peuvent être tirées pour les contraintes σ_y , τ_{xy} , τ_{xz} et τ_{yz} .

Les ingénieurs estiment souvent la précision de leur maillage à l'aide du calcul des $\Delta\sigma$. Le graphique 4.24 montre la relation entre l'écart des contraintes σ_x et l'erreur sur les arêtes calculée à l'aide de l'équation 4.2. On remarque que la procédure proposée pour l'estimation de l'erreur basée sur le Hessien du déplacement résultant permet une diminution des $\Delta\sigma_{\max}$. Ce résultat donne donc une signification plus physique aux calculs d'erreur qui ont été faits jusqu'à maintenant. Ainsi, une

diminution de l'erreur calculée à partir du Hessien du déplacement résultant diminue l'écart entre les contraintes pour les éléments voisins.

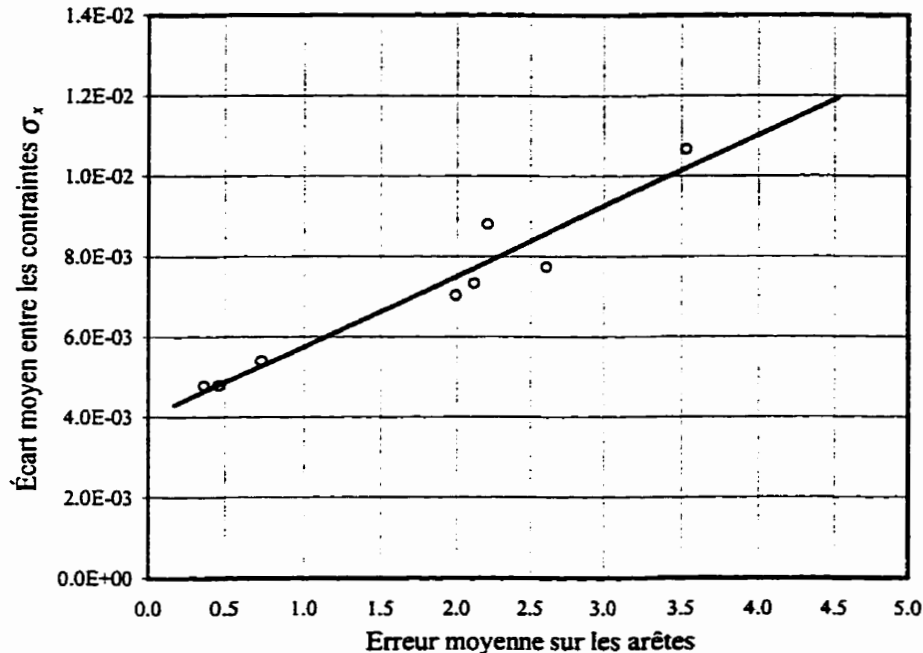


FIGURE 4.24: Relation entre l'erreur sur les arêtes et l'écart entre les contraintes

La section suivante présente les résultats obtenus pour une plaque carrée trouée en son centre. La solution théorique pour un tel problème est très complexe. L'erreur sera donc estimée à partir de l'écart entre les contraintes $\Delta\sigma_{\max}$.

4.5 Cas 5 : Plaque trouée orthotrope simplement supportée

4.5.1 Géométrie et propriétés du modèle

Dans cette section, nous examinerons le cas d'une plaque carrée trouée ayant des côtés de 1000 mm , soumise à une pression uniforme de 64.5 Pa . La figure 4.25 présente la géométrie du modèle analysé. La plaque à l'étude a une épaisseur de 6.8 mm

et est constituée du même stratifié que le cas 4. La plaque est trouée en son centre par un trou de 50 mm de rayon.

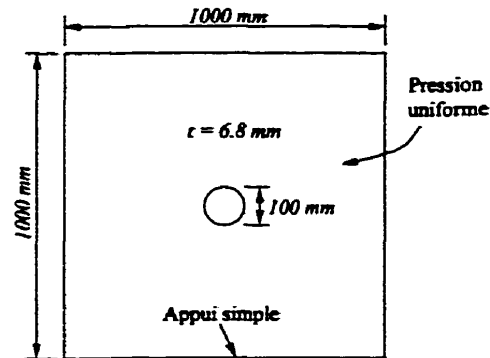


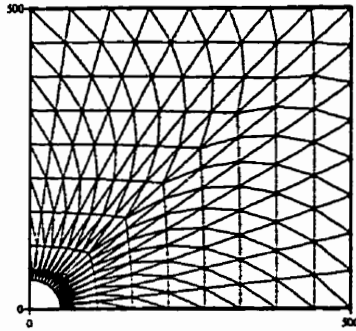
FIGURE 4.25: Description du modèle

Contrairement aux autres problèmes, la solution théorique pour le problème décrit à la figure 4.25 est très complexe. Par contre, nous avons démontré à la section 4.4.3 que l'écart des contraintes entre les éléments voisins est un bon indicateur de l'erreur sur le maillage. Dans cette section, nous utiliserons donc les résultats sur le calcul des contraintes afin de vérifier la qualité des maillages obtenus à la suite des processus d'adaptation.

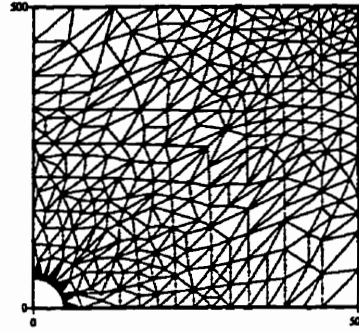
4.5.2 Résultats de l'adaptation de maillage

Les résultats qui vont suivre ont été obtenus en utilisant une longueur optimale dans la métrique de 0.08 mm^2 . Seulement la partie supérieure droite de la plaque a été maillée étant donné les conditions de symétrie. La figure 4.26 montre les maillages obtenus à partir du maillage initial pour les cinq premières itérations. Ces résultats ont été obtenus en prenant comme mesure de l'erreur la dérivée seconde du déplacement résultant (w) mesuré au plan moyen. Ces maillages ont été obtenus sans aucune restriction suivant la longueur minimum ou maximum permise des arêtes ou l'aspect géométrique des éléments.

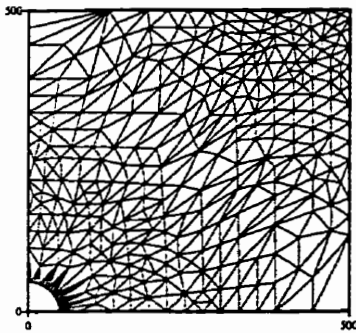
On remarque que les éléments tendent à se regrouper dans la région du centre de la plaque ainsi que près des coins. Étant donné que le problème analysé ici est peu différent du problème analysé à la section 4.4, il faut s'attendre à ce que les



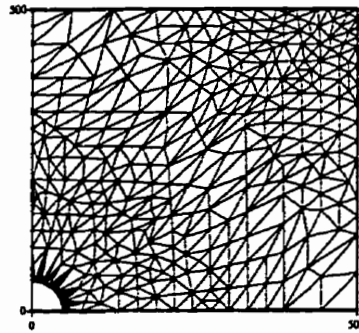
(a) Maillage initial



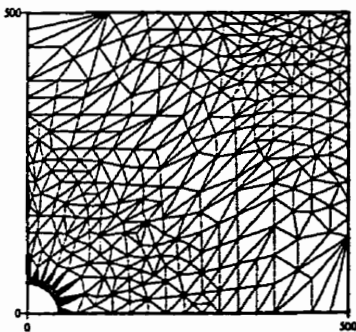
(b) Maillage après la première itération



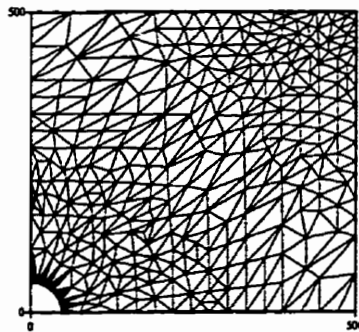
(c) Maillage après la deuxième itération



(d) Maillage après la troisième itération



(e) Maillage après la quatrième itération



(f) Maillage après la cinquième itération

FIGURE 4.26: Maillages obtenus au cours du processus d'adaptation

éléments se regroupent dans les mêmes régions étant donné que l'allure des dérivées secondes pour les deux problèmes sont très semblables sauf à proximité du trou.

Dans la région du trou, on pourrait s'attendre à une importante variation de la solution à cause des effets de concentration de contrainte et du fait que le moment passe d'une valeur maximale sur la plaque à nulle sur le contour du trou. On remarque par contre que les éléments tendent à être très allongés dans la radiale. Ceci peut être causé par le fait que les dérivées secondes ne peuvent être évaluées sur les frontières (voir section 3.1.2). Ainsi, les dérivées secondes autour du trou sont prises égales aux dérivées secondes calculées à l'intérieur du domaine à une distance non négligeable du trou. Il est donc très important de demeurer vigilant sur les résultats obtenus dans la zone du trou.

Aussi, on remarque que le nombre de nœuds sur la frontière définissant le trou est fixe au cours du processus d'itération. "Libom" définit la géométrie du domaine par des segments de droite seulement. Le quart de cercle définissant le trou est donc représenté par une série de segments de droite. Ainsi, les nœuds situés sur les intersections entre les segments de droite sont fixes puisqu'ils servent à définir la frontière du domaine. Le nombre de nœuds sur le contour du trou est donc supérieur ou égal au nombre qui serait normalement nécessaire puisque "Libom" ne peut enlever des nœuds mais il aurait pu, par contre, en ajouter.

Le tableau 4.13 présentent quelques résultats à la suite du calcul des contraintes au centre de chacun des éléments pour le maillage initial et le maillage obtenu à la suite de l'itération 4. Ces résultats démontrent l'efficacité du processus qui permet une uniformisation des contraintes sur le maillage. Bien sûr, nous ne pouvons pas conclure que les contraintes calculées convergent vers les contraintes exactes puisque celles-ci ne sont pas disponibles. En vertu des résultats obtenus au cours des analyses réalisées précédemment, on peut penser qu'ici aussi la solution numérique converge vers la solution exacte. De cette façon, les contraintes calculées numériquement convergeraient elles aussi vers les contraintes exactes.

Tous les résultats présentés jusqu'à maintenant ont été obtenus avec la même longueur optimale dans la métrique, soit une longueur de 0.08 mm^2 . Les résultats

TABLEAU 4.13: Statistiques sur le calcul des contraintes

		maillage figure 4.26(a)	maillage figure 4.26(e)
$\Delta\sigma_z$	Moyenne (MPa)	0.00637	0.00343
	Écart-type (MPa)	0.00690	0.00446
$\Delta\sigma_y$	Moyenne (MPa)	0.00637	0.00333
	Écart-type (MPa)	0.00690	0.00405
$\Delta\tau_{xy}$	Moyenne (MPa)	0.00280	0.00196
	Écart-type (MPa)	0.00301	0.00249
Nombre de nœuds		153	342

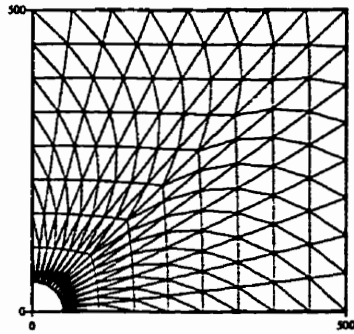
qui suivent ont été obtenus en faisant varier cette longueur afin de vérifier comment le nombre de nœuds obtenus pour les maillages adaptés varie en fonction du niveau choisi pour la longueur optimale dans la métrique.

Les figures 4.27(b), 4.27(c), 4.27(d) et 4.27(e) présentent les résultats obtenus pour des longueurs optimales dans la métrique de 0.15 mm^2 , 0.13 mm^2 , 0.10 mm^2 et 0.08 mm^2 respectivement. Le tableau 4.14 présente quelques résultats relatifs à ces maillages. Ces derniers sont les maillages obtenus à la suite d'un processus complet d'adaptation.

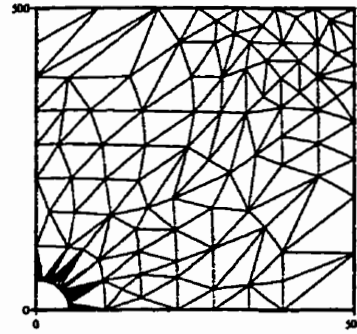
TABLEAU 4.14: Résultats vs longueur optimale dans la métrique

LoptMet (mm^2)	0.15	0.13	0.10	0.08
Moyenne des $\Delta\sigma_z$ (MPa)	0.1722	0.1437	0.1035	0.0802
Nombre de nœuds	113	160	233	344
Déplacement maximal (mm)	-1.1606	-1.1612	-1.1744	-1.1696

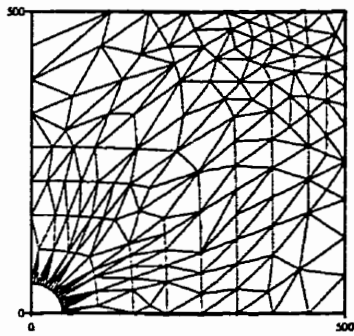
On voit sur la figure 4.27 que pour des choix différents de la longueur optimale dans la métrique, on obtient toujours le même motif d'éléments. Aussi, on remarque que plus la longueur optimale dans la métrique est petite, plus la densité d'éléments est élevée. Ainsi, pour un passage de la longueur optimale dans la métrique de 0.15 mm^2 à 0.08 mm^2 , on remarque une forte différence dans la densité des éléments. Le nombre de nœuds ou encore la densité du maillage est donc très influencée par la valeur du paramètre LoptMet.



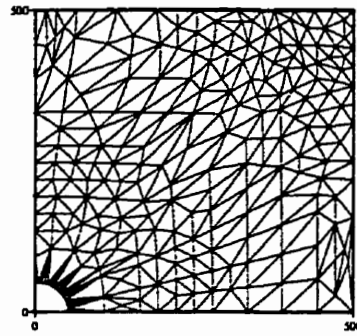
(a) Maillage uniforme



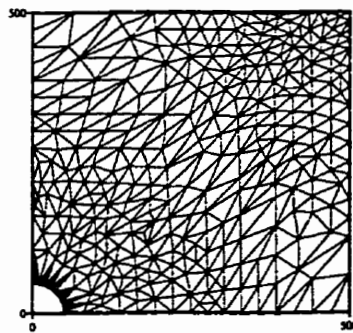
(b) Maillage adapté (niveau = 0.15)



(c) Maillage adapté (niveau = 0.13)



(d) Maillage adapté (niveau = 0.10)



(e) Maillage adapté (niveau = 0.08)

FIGURE 4.27: Maillages obtenus pour différente longueur optimale dans la métrique

4.5.3 Discussion

Afin de bien analyser les résultats obtenus, cette section présente quelques tableaux et graphiques permettant de mieux évaluer l'efficacité des maillages adaptés obtenus. Le graphique 4.28 nous montre les isocontours pour les composantes θ_x , θ_y et w du déplacement pour le maillage de la figure 4.26(f). On voit sur ce graphique que les maillages adaptés obtenus permettent d'obtenir une variation très lisse des différentes composantes du déplacement.

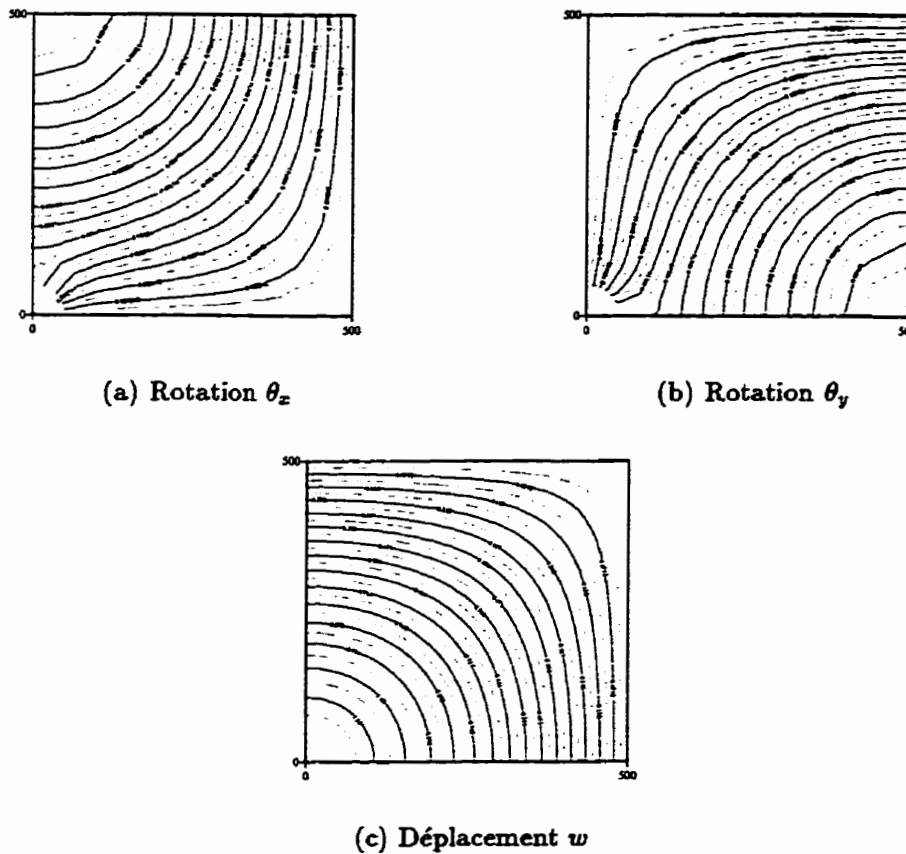


FIGURE 4.28: Isocontours pour les composantes θ_x , θ_y et w du déplacement

Le graphique 4.29 présente l'évolution de l'erreur sur les arêtes (Moyenne des $\Delta\sigma_{\max}$) en fonction du nombre de nœuds au cours d'un même processus d'adaptation. Il s'agit des résultats obtenus pour une longueur optimale dans la métrique de 0.08 mm^2 . Ce graphique démontre que le processus permet d'obtenir des solutions plus précises que celles obtenues des maillages structurés uniformément raffinés. Aussi, on remarque que le processus converge très rapidement vers le maillage final

convergé.

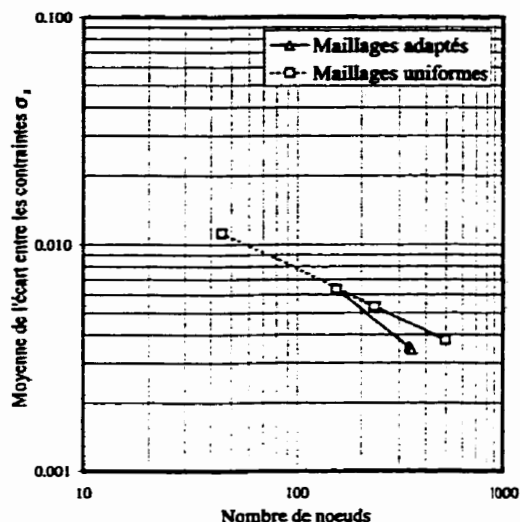


FIGURE 4.29: Erreur vs nombre de nœuds

Le graphique 4.30 présente les courbes de convergence pour les maillages uniformes et les maillages adaptés convergés. On peut voir sur ce graphique que le taux de convergence pour $\Delta\sigma_x$ est supérieur pour les maillages adaptés que pour les maillages uniformes. Cette constatation est un peu surprenante car le taux de convergence de l'écart entre les contraintes devrait normalement être le même puisque le taux de convergence est conditionné par l'élément utilisé et non le maillage [3, §4.3.6]. Ce point devrait donc faire l'objet d'une attention particulière lors de travaux futurs.

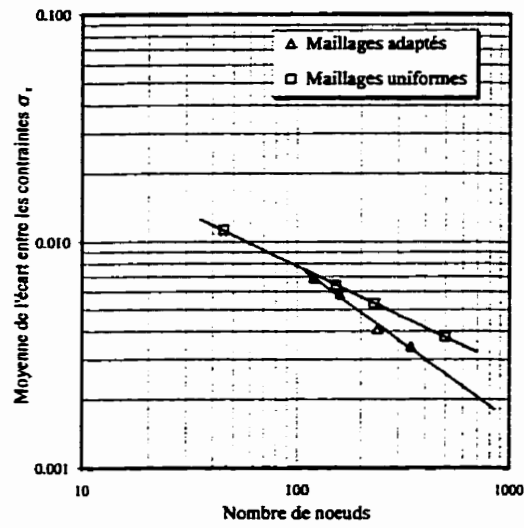


FIGURE 4.30: Précision des maillages uniformes et des maillages adaptés

Chapitre 5

Conclusion

5.1 Récapitulation et conclusion

Dans ce mémoire, plusieurs résultats numériques ont été présentés. Ceux-ci ont permis de bien évaluer les possibilités d'une stratégie d'adaptation de maillage basée sur une estimation de l'erreur calculée à partir des dérivées secondes de la solution obtenue par éléments finis. Ces résultats ont démontré l'efficacité de la méthode dans le cas des plaques fléchies.

Nous avons présenté, dans le deuxième chapitre, une description très détaillée de l'élément de type facette plane utilisé. Cet élément est composé de l'élément DST pour la partie flexion et de l'élément CST pour la partie membrane. Le calcul des matrices élémentaires pour chaque composante est effectué au niveau local alors que l'assemblage se fait au niveau d'un repère global. Les questions relatives à l'assemblage des parties flexion et membrane ont aussi été discutées.

Le troisième chapitre a porté essentiellement sur les bases théoriques de l'estimateur d'erreur utilisé. Nous avons présenté la théorie permettant d'utiliser, comme estimateur d'erreur, l'erreur d'interpolation intégrée sur une arête. Cette notion de l'erreur permet d'obtenir des informations à la fois quantitatives et directionnelles. Une description des moyens employés afin d'obtenir le maillage adapté a aussi été présentée.

Les résultats des analyses qui ont été réalisées sont présentés dans le quatrième chapitre. Plusieurs cas ont été présentés afin de démontrer que l'adaptation de maillage basé sur un estimateur d'erreur tel que celui présenté au chapitre trois donne des résultats très prometteurs dans un contexte de plaques fléchies.

Tout au long de la présentation des résultats, plusieurs détails permettant la validité des résultats obtenus et les moyens utilisés pour contourner certaines difficultés ont aussi été décrits. Ces difficultés étaient surtout dues au fait que l'erreur ne peut être calculée lorsqu'un point est situé sur une frontière du domaine. Dans un tel cas, c'est l'erreur du nœud interne voisin le plus près qui est utilisé. Malgré cela, les résultats obtenus ont été très concluants et les outils développés au cours de ces travaux se sont montrés très performants.

Le lecteur est invité à lire en annexe, la présentation des équations des différentes lois de comportement qui ont permis de créer les classes pour la gestion des matériaux (également présenté en annexe). Ainsi, dans le but d'obtenir un résumé complet des principales lois de comportement utilisées, les lois de comportement pour les matériaux isotropes, orthotropes et stratifiés y sont présentées. Également, une procédure prête à la programmation est présentée pour le calcul des coefficients de correction de cisaillement transverse intervenant dans l'évaluation de la loi de comportement des matériaux composites.

5.2 Recommandations et perfectionnements

Dans nos travaux, nous avons mis en évidence les difficultés rencontrés par le fait de négliger le terme intégral de surface lors de l'évaluation de l'équation 3.15 (page 43) pour le calcul du Hessien. En extrapolant les valeurs des nœuds internes voisins pour les nœuds situés sur des frontières, les problèmes de type poutre (très fréquent) où la solution varie suivant une seule direction ne peuvent être résolus sans l'intervention de limites sur la longueur des arêtes. Ce problème ne constitue pas une limitation grave mais il s'agit quand même d'un point négatif par rapport aux autres estimateurs qui peuvent correctement évaluer l'erreur sur les frontières. Il serait donc important de pouvoir évaluer ce terme de surface ou de développer une autre façon d'évaluer le Hessien.

Une autre difficulté mise en évidence par nos travaux découle du fait de décrire les frontières courbes par des segments de droite. En effet, nous avons vu lors de l'analyse de la plaque trouée que l'adaptation autour du trou était limitée par la dimension des segments de droite qui définissent la géométrie du trou. En permettant au logiciel d'adaptation de prendre en compte des frontières courbes, l'adaptation dans les zones courbes serait grandement améliorée.

Présentement, nous avons fait l'étude des problèmes de membrane et de flexion de façon séparée. Il serait intéressant de vérifier si l'utilisation du déplacement résultant donne de bons résultats dans le cas de chargements combinés, c'est-à-dire d'un chargement simultané en membrane et en flexion.

Finalement, nous n'avons réalisé que des plaques dans l'espace à deux dimensions. Ceci est dû au fait que "Libom" est limité aux domaines 2D. Il serait donc très important de développer rapidement des outils d'adaptation capables d'adapter des plaques orientées dans l'espace 3D. Avec de tels outils, beaucoup plus d'exemples pourraient être analysés. Ceci ouvrirait aussi la porte à l'étude de cas réels. On serait ainsi beaucoup plus en mesure de dire si la méthode à un avenir et si celle-ci se compare avantageusement à celles utilisées dans les logiciels commerciaux ou toute autre méthode.

Une partie de ces recommandations sont complètement ou partiellement réalisées dans le logiciel MEF++ développé par le GIREF. Un passage à MEF++ permettra donc très rapidement d'atteindre plusieurs de ces objectifs. Aussi, un passage à MEF++ faciliterait grandement les problèmes liés à "l'interfaçage" entre des logiciels écrits dans des langages différents.

Bibliographie

- [1] Abdala Filho J.E. et Dow J.O., *An error analysis approach for laminated composite plate finite element models* Computers and structures, Vol. 52, No. 4, pp. 611–616, 1994.
- [2] Barthelot J.M., *Matériaux composites, Comportement mécanique et analyse des structures* Masson, 2^e édition, 1996.
- [3] Bathe K.-J., *Finite element procedures* Prentice Hall, New Jersey, 1996.
- [4] Batoz J.L. et Lardeur P., *A discrete shear triangular element for the analysis of thick to very thin plates* Int. J. for numerical methods in engineering, Vol. 28, pp. 533–560, 1989.
- [5] Dhatt G. et Touzot G., *Une présentation de la méthode des éléments finis* Maloine S.A Éditeur, Paris, 1981.
- [6] Dhatt G. et Batoz J-L, *Modélisation des structures par éléments finis : solides élastiques* Les presses du l'Université Laval, Sainte-Foy, 1990.
- [7] Dhatt G. et Batoz J-L, *Modélisation des structures par éléments finis : poutres et plaques* Les presses du l'Université Laval, Sainte-Foy, 1990.
- [8] Dhatt G. et Batoz J-L, *Modélisation des structures par éléments finis : coques* Les presses du l'Université Laval, Sainte-Foy, 1992.
- [9] Diffpack group, *Diffpack reference manuel : basic tools* SINTEF, University of Oslo, Oslo, 1997.
- [10] Diffpack group, *Diffpack reference manuel : linear algebra tools* SINTEF, University of Oslo, Oslo, 1997.
- [11] Diffpack group, *Diffpack reference manuel : dpKernel, dpUtil and dpAppl* SINTEF, University of Oslo, Oslo, 1997.

-
- [12] Dompierre J., Valet M.-G., Fortin M., Habashi W.G., Aït-Ali-Yahia D., Boivin S., Bourgault Y. et Tam A. *Edge-Based mesh adaptive for CFD* Conf. on numerical methods for the Euler and Navier-Stokes Equations, Montréal, Septembre 14-16, pp. 265–299, 1995.
- [13] Habashi W.G. Dompierre J., Valet M.-G., Fortin M., Aït-Ali-Yahia D., Boivin S., Bourgault Y. et Tam A., *Anisotropic Mesh Optimization : Towards a Solver-Independent and Mesh-Intenpendent CFD*, VKI Lecture Series, 1996.
- [14] Hededal O., *Object-oriented structuring of finite elements* Departement of building technology and structural engineering, Aalborg university, Danemarque, 1994.
- [15] Horstmann C.S., *Mastering object-oriented design in C++* John Wiley and sons, New-York, 1995.
- [16] Labbé P. et Garon A., *A robust implementation of Zienkiewicz and Zhu's local patch recovery method* Comm. in numerical methods in engineering, Vol. 11, pp. 427–434, 1995.
- [17] Langtangen H.P., Pedersen G. et Shen W., *Finite element preprocessors in Diffpack* SINTEF, University of Oslo, Oslo, 1996.
- [18] Langtangen H.P. et Pedersen G., *Simple and flexible input data handling in C++ programs* SINTEF, University of Oslo, Oslo, 1996.
- [19] Langtangen H.P., *Getting started with finite element programming in Diffpack* SINTEF, University of Oslo, Oslo, 1994.
- [20] Langtangen H.P. et Nielsen B.F., *Getting started with Diffpack* SINTEF, University of Oslo, Oslo, 1995.
- [21] Langtangen H.P., *Diffpack : Software for partial differential equations* SINTEF, University of Oslo, Oslo, 1996.
- [22] Langtangen H.P., *Basic concepts in Diffpack* SINTEF, University of Oslo, Oslo, 1996.
- [23] Langtangen H.P., *Details of finite element programming in Diffpack* SINTEF, University of Oslo, Oslo, 1996.
- [24] Langtangen H.P., *A solver for the equations of linear thermo-elasticity* SINTEF, University of Oslo, Oslo, 1996.

- [25] Langtangen H.P., *A brief documentation of finite element fields over non-isoparametric elements in Diffpack* SINTEF, University of Oslo, Oslo, 1996.
- [26] Lardeur P. et Batoz J.L., *Composite plate analysis using a new discrete shear triangular finite element* Int. J. for numerical methods in engineering, Vol. 27, pp. 343–359, 1989.
- [27] Oh H.S. et Lim J.K., *Modified h-method with directional error estimate for finite element stress analysis* Computers and structures, Vol. 65, No. 2, pp. 191–204, 1997.
- [28] Samuelson A., Wiberg N.E. et Zeng L.F., *The effectivity of the Zienkiewicz-Zhu error estimate and two 2D adaptive mesh generators* Comm. in numerical methods in engineering, Vol. 9, pp. 687–699, 1993.
- [29] Tessler A., Riggs H.R., Colin E.F. et Geoffrey M.C., *An improved variational method for finite element stress recovery and a posteriori error estimation* Comp. methods in applied mechanics and engineering, Vol. 155, pp. 15–30, 1998.
- [30] Timoshenko S. et Woinowsky-Krieger S., *Theory of plates and shells* McGraw-Hill, 2^e édition, New-York, 1959.
- [31] Zienkiewicz O.C. et Zhu J.Z., *A simple error estimator and adaptive procedure for practice engineering analysis* Int. J. for numerical methods in engineering, Vol. 24, pp. 337–357, 1987.
- [32] Zienkiewicz O.C. et Zhu J.Z., *The superconvergent patch recovery and a posteriori error estimate. Part 1 : the recovery technique* Int. J. for numerical methods in engineering, Vol. 33, pp. 1331–1364, 1992.
- [33] Zienkiewicz O.C. et Zhu J.Z., *The superconvergent patch recovery and a posteriori error estimate. Part 2 : error estimate and adaptivity* Int. J. for numerical methods in engineering, Vol. 33, pp. 1365–1382, 1992.

Annexe A

Lois de comportement des plaques

À l'aide des bibliothèques "Diffpack" et du langage C++, nous avons créé des classes pour la gestion des matériaux composites qui exécutent les différentes tâches relatives à la création d'un laminé et au calcul des paramètres de sa loi de comportement. Ainsi, les principaux types de matériaux peuvent être créés. Jusqu'à maintenant, les classes permettant de créer un composite constitué de plis isotropes ou orthotropes ont été programmées.

Un laminé se définit comme étant un empilement de plis. Chaque pli se définit à l'aide d'une épaisseur et d'un angle dans le cas d'un matériau orthotrope. Quant au matériau constituant le pli, il se définit par l'ensemble des paramètres physiques qui le caractérisent (modules d'Young, coefficients de Poisson, etc.). On voit donc une hiérarchie se dessiner entre les matériaux de base et le laminé constitué par un empilement de ces matériaux de base. Ayant à l'esprit cette hiérarchisation, nous avons créé des classes qui respectent cette hiérarchie et qui permettent le calcul de toutes les informations nécessaires à l'évaluation de la loi de comportement dans le cas de l'élasticité linéaire.

Pour la présentation et la description des classes elles-mêmes, le lecteur peut se référer à l'annexe B (page 110). Les sections suivantes feront une brève description des informations qui devront être calculées pour l'évaluation des lois de comportement des matériaux.

A.1 Matériau isotrope

La particularité d'un matériau isotrope est que les caractéristiques physiques (ici mécaniques) sont constantes dans toutes les directions. Un matériau isotrope est défini à partir de deux paramètres physiques seulement : son module d'Young E et son coefficient de Poisson ν .

A.1.1 Élasticité tridimensionnelle

Pour l'élasticité tridimensionnelle, la loi de Hooke qui relie les contraintes et les déformations est définie de la façon suivante :

$$\{\sigma\} = [H]\{\epsilon\} \quad (\text{A.1})$$

avec

$$\langle \sigma \rangle = \langle \sigma_{xx} \quad \sigma_{yy} \quad \sigma_{zz} \quad \sigma_{xy} \quad \sigma_{xz} \quad \sigma_{yz} \rangle \quad (\text{A.2})$$

$$\langle \epsilon \rangle = \langle \epsilon_{xx} \quad \epsilon_{yy} \quad \epsilon_{zz} \quad \epsilon_{xy} \quad \epsilon_{xz} \quad \epsilon_{yz} \rangle \quad (\text{A.3})$$

et

$$[H] = \frac{E}{(1-2\nu)(1+\nu)} \begin{bmatrix} 1-\nu & \nu & \nu & 0 & 0 & 0 \\ \nu & 1-\nu & \nu & 0 & 0 & 0 \\ \nu & \nu & 1-\nu & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1-2\nu}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1-2\nu}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1-2\nu}{2} \end{bmatrix} \quad (\text{A.4})$$

A.1.2 Élasticité bidimensionnelle (hypothèse des contraintes planes)

L'hypothèse des contraintes planes (dans le plan xy) se traduit par :

$$\sigma_z = 0 \quad \epsilon_z \neq 0 \quad (\text{A.5})$$

L'hypothèse des contraintes planes est généralement admise dans le cas des structures minces de type poutre, plaque ou coque où z représente la direction suivant l'épaisseur. L'état de contrainte associé à un comportement de membrane et de flexion est de la forme :

$$\begin{aligned} \sigma_x &= \sigma_x(x, y, z) & \sigma_y &= \sigma_y(x, y, z) & \sigma_z &= 0 \\ \sigma_{xy} &= \sigma_{xy}(x, y, z) & \sigma_{xz} &= \sigma_{xz}(x, y, z) & \sigma_{yz} &= \sigma_{yz}(x, y, z) \end{aligned} \quad (\text{A.6})$$

La variation en z est linéaire pour les composantes σ_x , σ_y et σ_{xy} pour les matériaux isotropes. Pour définir la loi de comportement dans le cas des contraintes planes, on utilise l'inverse de la matrice $[H]$ du cas tridimensionnel (équation A.4) qui nous donne les déformations par rapport aux contraintes. Ainsi, en introduisant l'hypothèse $\sigma_z = 0$, on peut réduire la matrice $[H]^{-1}$ en supprimant la troisième ligne et la troisième colonne. En inversant de nouveau cette matrice réduite, on obtient la matrice $[H]$ pour l'élasticité bidimensionnelle avec l'hypothèse des contraintes planes. A noter que dans les relations qui vont suivre, la matrice $[H]$ a été séparée en deux matrices $[H]$ et $[H_\tau]$. Ceci est possible étant donné que la partie cisaillement transverse est découplée de la partie des efforts dans le plan de la plaque. La nouvelle matrice $[H]$ représente donc la matrice de comportement élastique pour les efforts dans le plan xy tandis que la matrice $[H_\tau]$ représente la matrice de comportement élastique pour les efforts tranchants transversaux agissant perpendiculairement au plan xy . La loi de Hooke pour un matériau isotrope bidimensionnel conforme à l'hypothèse des contraintes planes est donc :

$$\{\sigma\} = [H]\{\epsilon\} \quad \{\tau\} = [H_\tau]\{\gamma\} \quad (\text{A.7})$$

avec

$$\langle \sigma \rangle = \langle \sigma_x \quad \sigma_y \quad \tau_{xy} \rangle \quad \langle \tau \rangle = \langle \tau_{xz} \quad \tau_{yz} \rangle \quad (\text{A.8})$$

$$\langle \epsilon \rangle = \langle \epsilon_x \quad \epsilon_y \quad \gamma_{xy} \rangle \quad \langle \gamma \rangle = \langle \gamma_x \quad \gamma_y \rangle \quad (\text{A.9})$$

et

$$[H] = \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \quad [H_\tau] = \frac{E}{2(1+\nu)} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (\text{A.10})$$

aussi

$$\epsilon_z = -\frac{\nu}{E}(\sigma_x + \sigma_y) \quad (\text{A.11})$$

A.1.3 Élasticité bidimensionnelle (hypothèse des déformations planes)

En coordonnées cartésiennes, l'hypothèse classique des déformations planes se traduit par :

$$\epsilon_z = \gamma_{xz} = \gamma_{yz} = 0 \quad \sigma_z \neq 0 \quad (\text{A.12})$$

L'hypothèse des déformations planes est généralement admise lorsque la géométrie, les sollicitations, les conditions aux limites, les propriétés du matériaux sont suffisamment indépendantes de z : par exemple, dans un barrage dont les extrémités sont en butées ou dans un long tuyau sous pression interne sans possibilité de déformation dans le sens axial.

On obtient la loi de comportement d'un matériau bidimensionnel isotrope avec l'hypothèse des déformations planes de la même manière qu'on a obtenu la matrice de comportement avec l'hypothèse des contraintes planes : en introduisant l'hypothèse $\epsilon_z = 0$, on peut réduire la taille de la matrice de comportement tridimensionnelle en supprimant encore la troisième ligne et la troisième colonne. Ainsi, nous obtenons directement la matrice de comportement élastique avec l'hypothèse des déformations planes. La loi de Hooke pour un matériau isotrope bidimensionnel conforme à l'hypothèse des déformations planes est donc :

$$\{\sigma\} = [H]\{\epsilon\} \quad \{\tau\} = \{0\} \quad (\text{A.13})$$

avec

$$\langle \sigma \rangle = \langle \sigma_x \quad \sigma_y \quad \tau_{xy} \rangle \quad \langle \tau \rangle = \langle \tau_{xz} \quad \tau_{yz} \rangle \quad (\text{A.14})$$

$$\langle \epsilon \rangle = \langle \epsilon_x \quad \epsilon_y \quad \gamma_{xy} \rangle \quad (\text{A.15})$$

et

$$[H] = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & 0 \\ \nu & 1-\nu & 0 \\ 0 & 0 & \frac{1-2\nu}{2} \end{bmatrix} \quad (\text{A.16})$$

aussi

$$\sigma_z = \nu(\sigma_x + \sigma_y) \quad (\text{A.17})$$

A.2 Matériau orthotrope

Au contraire du matériau isotrope, le matériau orthotrope comporte des directions principales d'anisotropie orthogonales, c'est-à-dire que les caractéristiques physiques du matériau sont constantes selon trois directions orthogonales. Ainsi, dans le cas le plus général d'orthotropie, 9 constantes sont nécessaires pour définir la loi de comportement élastique : trois modules d'Young (E_L , E_T et E_z), trois coefficients de Poisson (ν_{TL} , ν_{Lz} et ν_{Tz}) et trois modules de cisaillement (G_{LT} , G_{Lz} , G_{Tz}). A noter que les indices représentent les directions selon le système d'axe présenté à la figure A.1. Les directions L , T et z sont parfois notées 1, 2 et 3 respectivement.

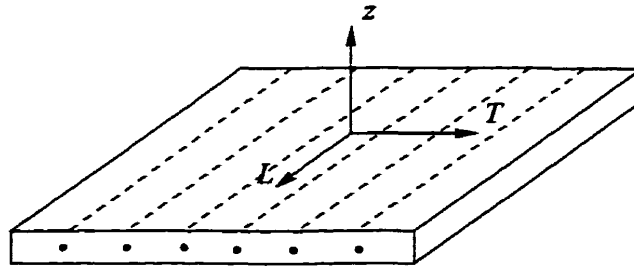


FIGURE A.1: Système d'axe local

Le plus souvent, seulement cinq coefficients sont nécessaires pour représenter un matériau orthotrope étant donné que dans la majorité des cas, le plan Tz est un plan isotrope. Cela signifie que peu importe l'orientation des axes T et z , les mêmes propriétés physiques sont mesurées. Par conséquent, nous pouvons écrire :

$$\begin{aligned} E_T = E_z \quad \nu_{LT} = \nu_{Lz} \quad G_{LT} = G_{Lz} \\ G_{Tz} = \frac{E_T}{2(1 + \nu_{Tz})} \quad \nu_{LT}E_T = \nu_{TL}E_L \end{aligned} \quad (\text{A.18})$$

Ces relations nous permettent de réduire le nombre de paramètres physiques nécessaires de 9 à 5. Maintenant, les cinq coefficients indépendants sont :

- deux modules d'Young (E_L, E_T);
- deux coefficients de Poisson (ν_{LT} ou ν_{TL} et ν_{Tz});
- un coefficient de cisaillement (G_{LT});

ou

- deux modules d'Young (E_L, E_T);
- un coefficient de Poisson (ν_{LT} ou ν_{TL});
- deux coefficients de cisaillement (G_{LT} ou G_{Lz} et G_{Tz}).

À noter qu'il est encore possible de diminuer le nombre de paramètres dans le cas d'un état de contraintes planes. Dans ce cas, le coefficient de poisson ν_{Tz} n'est pas requis et le nombre de paramètres physiques nécessaires est de 4.

A.2.1 Matériau orthotrope dans 3 directions

Comme il a été mentionné précédemment, dans le cas d'un matériau orthotrope dans 3 directions, la loi de Hooke se définit à partir des 9 paramètres physiques énoncés précédemment. Pour un matériau orthotrope, la loi de Hooke s'écrit de la façon suivante :

$$\{\sigma\} = [H]\{\epsilon\} \quad \Rightarrow \quad \{\epsilon\} = [H]^{-1}\{\sigma\} \quad (\text{A.19})$$

avec

$$\langle \sigma \rangle = \langle \sigma_{LL} \quad \sigma_{TT} \quad \sigma_{zz} \quad \tau_{LT} \quad \tau_{Lz} \quad \tau_{Tz} \rangle \quad (\text{A.20})$$

$$\langle \epsilon \rangle = \langle \epsilon_{LL} \quad \epsilon_{TT} \quad \epsilon_{zz} \quad \gamma_{LT} \quad \gamma_{Lz} \quad \gamma_{Tz} \rangle \quad (\text{A.21})$$

et

$$[H]^{-1} = \begin{bmatrix} \frac{1}{E_L} & -\frac{\nu_{TL}}{E_T} & -\frac{\nu_{zL}}{E_z} & 0 & 0 & 0 \\ -\frac{\nu_{LT}}{E_T} & \frac{1}{E_T} & -\frac{\nu_{zT}}{E_z} & 0 & 0 & 0 \\ -\frac{\nu_{Lz}}{E_L} & -\frac{\nu_{TL}}{E_z} & \frac{1}{E_z} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{G_{LT}} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{G_{Lz}} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{G_{Tz}} \end{bmatrix} \quad (\text{A.22})$$

En inversant l'équation A.22, on obtient l'expression de la matrice de comportement pour le cas d'un matériau orthotrope dans 3 directions orthogonales.

$$[H] = \begin{bmatrix} \frac{(\nu_{Tz}\nu_{zT}-1)E_L}{a} & -\frac{(\nu_{Tz}\nu_{zL}+\nu_{TL})E_L}{a} & -\frac{(\nu_{TL}\nu_{zT}+\nu_{zL})E_L}{a} & 0 & 0 & 0 \\ -\frac{(\nu_{Lz}\nu_{zT}+\nu_{LT})E_T}{a} & -\frac{(\nu_{Lz}\nu_{zL}-1)E_T}{a} & -\frac{(\nu_{LT}\nu_{zL}+\nu_{zL})E_T}{a} & 0 & 0 & 0 \\ -\frac{(\nu_{LT}\nu_{Tz}+\nu_{Lz})E_z}{a} & -\frac{(\nu_{Lz}\nu_{TL}+\nu_{TL})E_z}{a} & -\frac{(\nu_{LT}\nu_{TL}-1)E_z}{a} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{G_{LT}} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{G_{Lz}} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{G_{Tz}} \end{bmatrix} \quad (\text{A.23})$$

avec

$$a = \nu_{Tz}\nu_{zT} + \nu_{LT}\nu_{TL} + \nu_{LT}\nu_{Tz}\nu_{zL} + \nu_{Lz}\nu_{TL}\nu_{zT} + \nu_{Lz}\nu_{zL} - 1 \quad (\text{A.24})$$

Pour respecter les conditions de symétrie et pour s'assurer que la matrice $[H]^{-1}$ soit définie positive, les relations suivantes doivent être respectées.

$$\frac{\nu_{LT}}{E_L} = \frac{\nu_{TL}}{E_T} \quad \frac{\nu_{Lz}}{E_L} = \frac{\nu_{zL}}{E_z} \quad \frac{\nu_{Tz}}{E_T} = \frac{\nu_{zT}}{E_z} \quad (\text{A.25})$$

$$\begin{aligned} 1 - \nu_{LT}\nu_{TL} &> 0 & 1 - \nu_{Lz}\nu_{zL} &> 0 & 1 - \nu_{Tz}\nu_{zT} &> 0 \\ 1 - \nu_{LT}\nu_{TL} - \nu_{Lz}\nu_{zL} - \nu_{Tz}\nu_{zT} - \nu_{LT}\nu_{Tz}\nu_{zL} - \nu_{TL}\nu_{Lz}\nu_{zT} &> 0 \end{aligned} \quad (\text{A.26})$$

A.2.2 Matériau orthotrope dans le plan LT et isotrope dans le plan Tz

Dans un tel cas, la loi de Hooke est fonction de 5 coefficients physiques seulement. Ici, la loi de Hooke nous donne la relation entre les contraintes par rapport au système

d'axe local et les déformations également définies par rapport au système d'axe local. Les relations sont les suivantes :

$$\{\sigma_L\} = [H_L]\{\epsilon_L\} \quad \{\tau_L\} = [H_{\tau L}]\{\gamma_L\} \quad (\text{A.27})$$

avec

$$\langle \sigma_L \rangle = \langle \sigma_L \quad \sigma_T \quad \tau_{LT} \rangle \quad \langle \tau_L \rangle = \langle \tau_{Lz} \quad \tau_{Tz} \rangle \quad (\text{A.28})$$

$$\langle \epsilon_L \rangle = \langle \epsilon_L \quad \epsilon_T \quad \gamma_{LT} \rangle \quad \langle \gamma_L \rangle = \langle \gamma_{Lz} \quad \gamma_{Tz} \rangle \quad (\text{A.29})$$

et

$$[H_L] = \begin{bmatrix} \frac{E_L}{1-\nu_{LT}\nu_{TL}} & \frac{E_T\nu_{LT}}{1-\nu_{LT}\nu_{TL}} & 0 \\ \frac{E_T\nu_{LT}}{1-\nu_{LT}\nu_{TL}} & \frac{E_T}{1-\nu_{LT}\nu_{TL}} & 0 \\ 0 & 0 & \frac{1}{G_{LT}} \end{bmatrix} \quad (\text{A.30})$$

Étant donné que le plan Tz est un plan isotrope, Les relations A.18 existant entre les paramètres physiques sont toujours valables.

A.2.3 Rotations des contraintes et des déformations

Il est possible de *tourner* les contraintes et les déformations calculées précédemment pour les exprimer dans un autre système d'axe. Ainsi, tel que nous le verrons plus loin, les matériaux orthotropes sont le plus souvent empilés les uns sur les autres dans des orientations différentes pour former un stratifié (aussi appelé laminé).

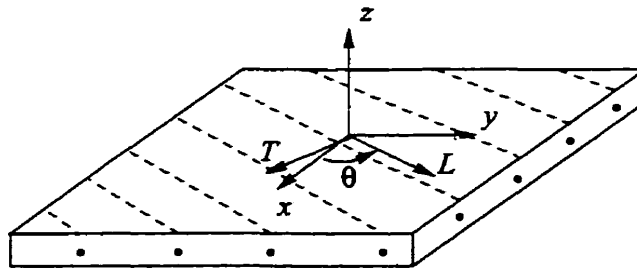


FIGURE A.2: Système d'axe local et global

Pour calculer les caractéristiques physiques du stratifié, nous devons mesurer les contraintes et les déformations de chaque couche par rapport à un même système

d'axe global (sur la figure A.2, il s'agit du système xyz). Les relations permettant de *tourner* les contraintes et les déformations du système d'axe local LTz vers le système global xyz sont :

$$\{\sigma\} = [H]\{\epsilon\} \quad \{\tau\} = [H_\tau]\{\gamma\} \quad (\text{A.31})$$

avec

$$\langle \sigma \rangle = \langle \sigma_x \quad \sigma_y \quad \tau_{xy} \rangle \quad \langle \tau \rangle = \langle \tau_{xz} \quad \tau_{yz} \rangle \quad (\text{A.32})$$

$$\langle \epsilon \rangle = \langle \epsilon_x \quad \epsilon_y \quad \gamma_{xy} \rangle \quad \langle \gamma \rangle = \langle \gamma_{xz} \quad \gamma_{yz} \rangle \quad (\text{A.33})$$

et

$$[H] = [T_1]^T [H_L] [T_1] \quad [H_\tau] = [T_2]^T [H_{\tau L}] [T_2] \quad (\text{A.34})$$

$$[T_1] = \begin{bmatrix} \cos^2 \theta & \sin^2 \theta & \cos \theta \sin \theta \\ \sin^2 \theta & \cos^2 \theta & -\cos \theta \sin \theta \\ -2 \cos \theta \sin \theta & 2 \cos \theta \sin \theta & \cos^2 \theta - \sin^2 \theta \end{bmatrix} \quad (\text{A.35})$$

$$[T_2] = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \quad (\text{A.36})$$

On pourrait aussi démontrer à l'aide des équations A.31 et A.34 que :

$$\{\epsilon_L\} = [T_1]\{\epsilon\} \quad \{\sigma\} = [T_1]^T \{\sigma_L\} \quad (\text{A.37})$$

$$\{\gamma_L\} = [T_2]\{\gamma\} \quad \{\tau\} = [T_2]^T \{\tau_L\} \quad (\text{A.38})$$

Ainsi, à l'aide de toutes ces relations, il nous sera possible de définir un stratifié composé d'un empilement de couches orthotropes tel qu'illustré sur la figure A.3 :

A.3 Matériau composite

Un matériau composite, aussi appelé stratifié ou laminé, est obtenu à partir d'un empilement de couches. Ces couches, généralement composées de matériaux

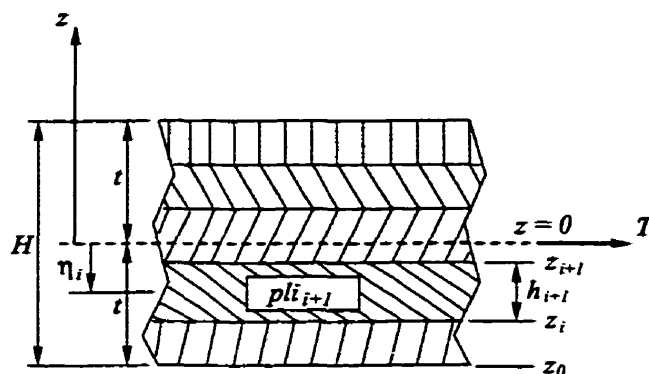


FIGURE A.3: Variables utilisées pour les calculs

orthotropes, sont appelées ici *plis*. Un pli est défini par son épaisseur et par un angle que font les fibres par rapport à une direction de référence. L'angle d'un pli θ est mesuré entre l'axe global x et l'axe L . L'angle est de plus mesuré dans le sens anti-horaire (voir figure A.2).

Pour définir un laminé, on considère un empilement de couches isotropes ou orthotropes dont les axes d'orthotropie sont L , T et z avec isotropie d'axe L (dans le plan Tz). La description d'un laminé a travers son épaisseur est la suivante :

- chaque couche i est définie par les plans $z = z_i$ et $z = z_{i+1}$ et $z_i \leq z \leq z_{i+1}$;
- les directions d'orthotropie L et T pour chaque couche i sont représentées par l'angle θ_i ;
- la loi de comportement de chaque pli est basée sur l'hypothèse des contraintes planes ;
- la continuité cinématique est admise entre les plis (les plis sont parfaitement collés et l'épaisseur de la colle est négligeable).

Ces hypothèses nous permettent d'écrire les relations entre les contraintes et les déformations pour chaque couche i de façon découplées. Ainsi pour chaque couche i , les relations données à la section A.2 peuvent être calculées séparément.

Il est possible de calculer pour une plaque ou une coque fabriquée d'un matériau composite les efforts résultants N , T et M qui représentent les efforts normaux, les efforts tranchants et les moments fléchissants respectivement. Ces efforts sont définis ainsi :

$$\{N\} = \begin{Bmatrix} N_x \\ N_y \\ N_{xy} \end{Bmatrix} = \int_{-t}^{+t} \begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} dz = [H_m] \{e\} + [H_{mf}] \{\kappa\} \quad (\text{A.39})$$

$$\{M\} = \begin{Bmatrix} M_x \\ M_y \\ M_{xy} \end{Bmatrix} = \int_{-t}^{+t} \begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} z dz = [H_{mf}] \{e\} + [H_f] \{\kappa\} \quad (\text{A.40})$$

$$\{T\} = \begin{Bmatrix} T_x \\ T_y \end{Bmatrix} = \int_{-t}^{+t} \begin{Bmatrix} \tau_{xz} \\ \tau_{yz} \end{Bmatrix} dz = [H_c] \{\gamma\} \quad (\text{A.41})$$

Les vecteurs $\{e\}$, $\{\kappa\}$ sont respectivement les vecteurs des déformations de membrane et de flexion. Le vecteur $\{\gamma\}$ est le vecteur des déformations de cisaillement transverse. La matrice $[H_m]$ est la matrice des déformations de membrane, la matrice $[H_f]$ est la matrice des déformations de flexion, la matrice $[H_c]$ est la matrice des déformations de cisaillement et la matrice $[H_{mf}]$ est la matrice du couplage entre les déformations de membrane et de flexion. Pour un composite possédant une symétrie matérielle par rapport au plan $z = 0$, il n'y a pas de couplage entre les effets de membrane et de flexion, la matrice $[H_{mf}]$ est donc nulle. Dans ce cas, le plan xy est alors le plan neutre : des efforts de membranes appliqués dans le plan neutre ne produisent aucune courbure et, réciproquement, des efforts de flexion ne produisent aucune déformation de membrane. Les matrices $[H_m]$, $[H_f]$, $[H_{mf}]$ et $[H_c]$ sont définies de la façon suivante :

$$[H_m] = \int_{-t}^{+t} [H] dz \quad (\text{A.42})$$

$$[H_{mf}] = \int_{-t}^{+t} [H] z dz \quad (\text{A.43})$$

$$[H_f] = \int_{-t}^{+t} [H] z^2 dz \quad (\text{A.44})$$

Évidemment, dans les équations A.42, A.43 et A.44, la matrice $[H]$ représente la matrice de comportement du pli tel que défini dans les équations A.10 ou A.30.

Pour une plaque composée de N couches orthotropes ou isotropes (dont les caractéristiques sont constantes par couche), les matrices précédentes peuvent s'écrire ainsi :

$$[H_m] = \sum_{i=1}^N h_i [H]_i \quad (\text{A.45})$$

$$[H_{mf}] = \sum_{i=1}^N h_i \eta_i [H]_i \quad (\text{A.46})$$

$$[H_f] = \sum_{i=1}^N \frac{1}{3} (z_{i+1}^3 - z_i^3) [H]_i \quad (\text{A.47})$$

avec

$$h_i = z_{i+1} - z_i \quad \eta_i = \frac{1}{2}(z_{i+1} + z_i) \quad (\text{A.48})$$

A.3.1 Calcul de la matrice $[H_c]$

Le calcul de la matrice $[H_c]$ peut être basé sur des considérations d'équilibre statique et d'équivalence énergétique de manière à ce que la rigidité en cisaillement transversale CT du modèle de plaque corresponde le plus possible à celle définie par la théorie de l'élasticité tridimensionnelle. Ces considérations sont en accord avec celle de la théorie mixte des plaques de Reissner et conduit pour les plaques isotrope à un facteur de correction de cisaillement égale à 5/6.

La démarche présentée ci-dessous pour les plaques stratifiées admet qu'il n'y a pas de couplage membrane-flexion, ainsi, la matrice $[H_{mf}] = [0]$. La matrice $[H_c]$ est définie de sorte que la densité surfacique d'énergie interne de CT obtenue pour une distribution tridimensionnelle (exacte) des contraintes σ_{zz} et σ_{yz} , notée U_1 , soit identique à celle, notée U_2 , associé au modèle bidimensionnel basée sur les hypothèses de plaques Reissner/Mindlin.

$$U_1 = \frac{1}{2} \int_{-t}^{+t} \langle \tau \rangle [H_\tau]^{-1} \{ \tau \} dz \quad (\text{A.49})$$

$$U_2 = \frac{1}{2} \langle T \rangle [H_c]^{-1} \{ T \} = \frac{1}{2} \langle \gamma \rangle [H_c] \{ \gamma \} \quad (\text{A.50})$$

À l'aide de l'équation A.41, on obtient :

$$U_2 = \frac{1}{2} \left(\int_{-t}^{+t} \langle \tau \rangle dz \right) [H_c]^{-1} \left(\int_{-t}^{+t} \{ \tau \} dz \right) \quad (\text{A.51})$$

Pour obtenir $[H_c]$ il faut ainsi connaître la distribution de $\tau(z)$. Celle-ci est obtenue en considérant les équations d'équilibre homogène suivant x et y :

$$\begin{aligned} \tau_{zz}(z) &= - \int_{-t}^z (\sigma_{zz,z} + \tau_{zy,y}) dz \\ \tau_{zz}(z) &= - \int_{-t}^z (\tau_{zy,z} + \sigma_{yy,y}) dz \end{aligned} \quad (\text{A.52})$$

avec

$$\tau_{xz} = \tau_{yz} = 0 \quad \text{pour } z = \pm t \quad (\text{A.53})$$

Les contraintes planes σ_{zz} , σ_{yy} et τ_{zy} s'expriment en fonction des efforts résultants. Ainsi, en considérant $[H_{mf}] = [0]$, on obtient :

$$\begin{Bmatrix} \sigma_{zz} \\ \sigma_{yy} \\ \tau_{zy} \end{Bmatrix} = \{ \sigma(z) \} = [H(z)] ([H_m]^{-1} \{ N \} + z [H_f]^{-1} \{ M \}) \quad (\text{A.54})$$

Dans le cas d'un comportement de flexion/cisaillement pur ($\{ N \} = \{ 0 \}$), on obtient :

$$\{ \sigma(z) \} = z [A(z)] \{ M \} \quad (\text{A.55})$$

avec

$$[A(z)] = [H(z)] [H_f]^{-1} \quad (\text{A.56})$$

Pour déterminer $[H_c]$ on considère que $[H(z)]$ et $[H_f]$ ne dépendent pas de x et de y . Les équations A.55 et A.56 nous conduisent alors à :

$$\begin{Bmatrix} \tau_{zz} \\ \tau_{yz} \end{Bmatrix} = \{ \tau(z) \} = [D_1(z)] \{ T \} + [D_2(z)] \{ \lambda \} \quad (\text{A.57})$$

avec

$$[D_1(z)] = \int_{-t}^z -\frac{z}{2} \begin{bmatrix} A_{11} + A_{33} & A_{13} + A_{32} \\ A_{31} + A_{23} & A_{22} + A_{33} \end{bmatrix} dz \quad (\text{A.58})$$

$$[D_2(z)] = \int_{-t}^z -\frac{z}{2} \begin{bmatrix} A_{11} - A_{33} & A_{13} - A_{32} & 2A_{12} & 2A_{31} \\ A_{31} - A_{23} & A_{33} - A_{22} & 2A_{32} & 2A_{21} \end{bmatrix} dz \quad (\text{A.59})$$

et

$$\langle T \rangle = \langle T_x \quad T_y \rangle = \langle M_{x,x} + M_{xy,y} \quad M_{xy,x} + M_{y,y} \rangle \quad (\text{A.60})$$

$$\langle \lambda \rangle = \langle M_{x,x} - M_{xy,y} \quad M_{xy,x} - M_{y,y} \quad M_{y,x} \quad M_{x,y} \rangle \quad (\text{A.61})$$

En tenant compte de toutes ces relations, U_1 peut maintenant s'écrire de la façon suivante :

$$U_1 = \frac{1}{2} \langle T \quad \lambda \rangle \begin{bmatrix} [C_{11}] & [C_{12}] \\ [C_{12}]^T & [C_{22}] \end{bmatrix} \begin{Bmatrix} T \\ \lambda \end{Bmatrix} \quad (\text{A.62})$$

avec

$$[C_{11}]_{2 \times 2} = \int_{-t}^{+t} [D_1]^T [H_r]^{-1} [D_1] dz \quad (\text{A.63})$$

$$[C_{12}]_{2 \times 4} = \int_{-t}^{+t} [D_1]^T [H_r]^{-1} [D_2] dz \quad (\text{A.64})$$

$$[C_{22}]_{4 \times 4} = \int_{-t}^{+t} [D_2]^T [H_r]^{-2} [D_1] dz \quad (\text{A.65})$$

L'équivalence $U_1 = U_2$ permettant de définir la matrice $[H_c]$ conduit finalement à :

$$\langle T \quad \lambda \rangle \begin{bmatrix} [C_{11}] - [H_c]^{-1} & [C_{12}] \\ [C_{12}]^T & [C_{22}] \end{bmatrix} \begin{Bmatrix} T \\ \lambda \end{Bmatrix} = 0 \quad \forall \{T\}, \{\lambda\} \quad (\text{A.66})$$

Cette relation matricielle nous amène à proposer pour toutes les situations de stratification que :

$$[H_c] = [C_{11}]^{-1} \quad (\text{A.67})$$

En comparant $[H_c]$ ainsi calculé avec $[\tilde{H}_c] = \int_{-t}^{+t} [H_r] dz$, on peut calculé les coefficients de correction de cisaillement :

$$k_1 = \frac{(H_c)_{11}}{(\tilde{H})_{11}} \quad k_{12} = \frac{(H_c)_{12}}{(\tilde{H})_{12}} \quad k_2 = \frac{(H_c)_{22}}{(\tilde{H})_{22}} \quad (\text{A.68})$$

Pour un composite composée de N plis, on peut écrire les relations précédentes permettant de calculer la matrice $[C_{11}]$ qui va nous conduire à la matrice $[H_c]$. Ainsi, pour chaque couche i on a :

$$[D_{1i}(z)] = \sum_{j=1}^i \int_{z_{j-1}}^{z_j} -\frac{z}{2} \begin{bmatrix} A_{11} + A_{33} & A_{13} + A_{32} \\ A_{31} + A_{23} & A_{22} + A_{33} \end{bmatrix}_j dz \quad (\text{A.69})$$

$$\begin{aligned} [D_{1i}(z)] = & \sum_{j=1}^{i-1} \left(-\frac{1}{4} \begin{bmatrix} A_{11} + A_{33} & A_{13} + A_{32} \\ A_{31} + A_{23} & A_{22} + A_{33} \end{bmatrix}_j (z_j^2 - z_{j-1}^2) \right) - \\ & -\frac{1}{4} \begin{bmatrix} A_{11} + A_{33} & A_{13} + A_{32} \\ A_{31} + A_{23} & A_{22} + A_{33} \end{bmatrix}_i (z_i^2) + \\ & -\frac{1}{4} \begin{bmatrix} A_{11} + A_{33} & A_{13} + A_{32} \\ A_{31} + A_{23} & A_{22} + A_{33} \end{bmatrix}_i z^2 \end{aligned} \quad (\text{A.70})$$

en prenant

$$[\tilde{C}_{1i}] = -\frac{1}{4} \begin{bmatrix} A_{11} + A_{33} & A_{13} + A_{32} \\ A_{31} + A_{23} & A_{22} + A_{33} \end{bmatrix} \quad (\text{A.71})$$

on obtient ainsi :

$$[D_{1i}(z)] = \underbrace{\sum_{j=1}^{i-1} \left([\tilde{C}_{1i}] (z_j^2 - z_{j-1}^2) \right)}_{[M_i]} - [\tilde{C}_{1i}] z^2 + [\tilde{C}_{1i}] z^2 \quad (\text{A.72})$$

et

$$[D_{1i}(z)]^T = \left[\sum_{j=1}^{i-1} \left([\tilde{C}_{1i}]^T (z_j^2 - z_{j-1}^2) \right) - [\tilde{C}_{1i}]^T z^2 \right] + [\tilde{C}_{1i}]^T z^2 \quad (\text{A.73})$$

L'équation A.63 pour le calcul de la matrice $[C_{11}]$ devient finalement :

$$[C_{11}]_{2 \times 2} = \int_{-t}^{+t} [D_1(z)]^T [H_r(z)]^{-1} [D_1(z)] dz \quad (\text{A.74})$$

$$\Rightarrow [C_{11}]_{2 \times 2} = \sum_{i=1}^N \int_{z_{i-1}}^{z_i} [D_{1i}(z)]^T [H_{ri}]^{-1} [D_{1i}(z)] dz \quad (\text{A.75})$$

d'où finalement :

$$[C_{11}]_{2 \times 2} = \sum_{i=1}^N \int_{z_{i-1}}^{z_i} \left([M_i]^T + [\bar{C}_{1i}]^T z^2 \right) [H_{ri}]^{-1} \left([M_i] + [\bar{C}_{1i}] z^2 \right) dz \quad (\text{A.76})$$

Ainsi, à partir de la dernière équation, il nous est possible de calculer explicitement la matrice $[C_{11}]$ en faisant la multiplication et l'intégration de l'équation A.76 sur un logiciel de calcul symbolique (*Maple V*). Ainsi, après avoir calculé la matrice $[C_{11}]$ à l'aide des équations A.71, A.72, A.73 et A.76, il ne nous reste plus qu'à inverser cette matrice pour obtenir la matrice $[H_c]$ et les différents facteurs de correction de cisaillement peuvent être calculés à l'aide des relations de l'équation A.68 .

Annexe B

Description des classes pour la gestion des matériaux

À partir des différentes équations présentées à la section précédente, nous avons créé des classes pour la gestion des matériaux composites. Ces classes permettent de définir les paramètres de rigidité (E , ν , G ,...) de chaque matériau utilisé dans un modèle. Elles permettent aussi de calculer la loi de comportement de chaque matériau. Elles sont aussi responsables de la gestion de l'empilement des plis pour former un laminé. Chaque pli peut être constitué d'un matériau isotrope ou orthotrope.

Les classes réalisées, avec le langage C++, utilisent la notion de polymorphisme (fonction virtuelle) dans la hiérarchisation des classes permettant la création d'un laminé. Ainsi, certaines fonctions doivent nécessairement apparaître dans les classes dérivées pour conserver la bonne utilisation de l'ensemble des classes. On peut voir sur la figure B.1 la hiérarchisation des classes ainsi que les fonctions virtuelles pures constituant chacune d'elles. Rappelons qu'une fonction virtuelle pure est une fonction qui doit nécessairement apparaître dans les classes dérivées et qu'aucun objet d'une classe contenant une fonction virtuelle pure ne peut être créé. Ceci définit ainsi une interface qui devra être respectée en tout temps lors de la création de nouvelles classes.

Maintenant voici une description plus détaillée de chaque classe ainsi que des fonctions et données membres constituant chacune d'elles :

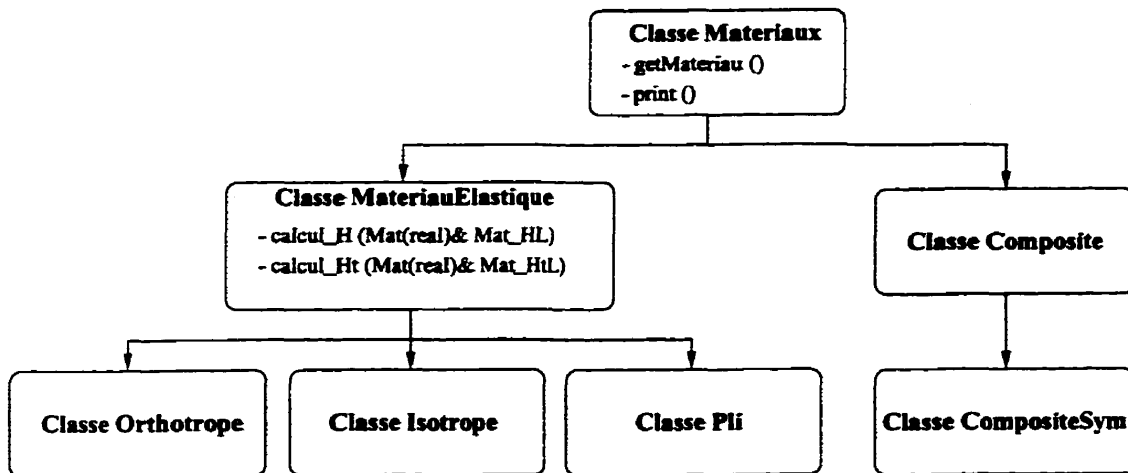


FIGURE B.1: Hiérarchie des classes pour la gestion des matériaux

B.1 Classe Materiaux et MateriauxElastique

```

class Materiaux
{
public:
    virtual String Getmateriau() = 0;
    virtual print() = 0;
};

class Materiauxelastique : public Materiaux
{
public:
    virtual Calcul_H(Mat(real)& Mat_HL) = 0 ;
    virtual Calcul_Ht(Mat(real)& Mat_HtL) = 0;
};
  
```

La classe **Materiaux** constitue la base de toutes les classes dérivées pour la gestion des matériaux composites. Elle comporte 2 fonctions virtuelles :

virtual String Getmateriau() Cette fonction retourne le nom du matériau créé.

Ceci suppose donc que chaque matériau créé comporte un nom de type String qui peut être obtenu en tout temps par cette fonction.

virtual print() Cette fonction permet l'affichage de différents paramètres comme le module d'Young, le coefficient de Poisson, le nombres de plis, etc. d'un matériau.

La classe `Materiauxelastique` est utilisée pour définir des matériaux de type élastique, c'est-à-dire que la relation contrainte-déformation pour ces matériaux est définie à partir des matrices $[H]$ et $[H_{\tau}]$ du même type que celles présentées aux sections A.1 et A.2. Ainsi, chaque classe dérivée de la classe `Materiauxelastique` (`Isotrope`, `Orthotrope` et `Pli`) doit contenir les fonctions virtuelles suivantes :

`virtual Calcul_H(...)` Comme il a été mentionné au paragraphe précédent, cette fonction retourne la matrice $[H]$ pour le calcul des contraintes en fonction des déformations par l'intermédiaire de la matrice `Mat_HL` passée en argument.

`virtual Calcul_Ht(...)` Cette fonction, comme la précédente, retourne une matrice par l'intermédiaire de la matrice `Mat_HtL` passée en argument. La matrice qui est retournée est la matrice $[H_{\tau}]$ servant dans le calcul des contraintes de cisaillement. (voir équation A.10, A.16 et A.23)

B.2 Classe Isotrope

```
class Isotrope : public Materiauxelastique
{
private:
    String    nom;           // Nom du materiau
    Boolean    Hcalcule;     // Vrai si la matrice [H] est calcule
    Boolean    Htcalcule;   // Vrai si la matrice [Ht] est calcule
    real      E;            // Module d'young E
    real      nu;           // Coefficient de poisson nu
    Mat(real) mat_HL;       // Matrice [H longitudinal]
    Mat(real) mat_HtL;      // Matrice [Ht] = [Htau]
    char      *Type;        // Type du materiau

public:
    Isotrope(real Etmp, real nutmp, String name);
    Isotrope();
    ~Isotrope() {};

    Modifier_Par_Phys (real Etmp, real nutmp, String name);

    Calcul_H(Mat(real)& Mat_HL);
```

```

Calcul_Ht(Mat(real)& Mat_HtL);

String Getmateriau() { return nom;}
print();
};

```

La classe **Isotrope** dérivée de la classe **Materiauxelastique** permet, comme son nom l'indique, de créer un matériau de type isotrope qui pourra ou non être inclus dans un empilement de pli pour créer un laminé (voir section A.1). Cette classe comporte plusieurs données membres privées. Les données membres n'ont pas besoin de plus d'explication étant donné qu'ils sont brièvement décrits dans le code.

Comme cette classe est dérivée de la classe **Materiauxelastique** et naturellement de la classe **Materiau**, elle contient les fonctions virtuelles énumérées et expliquées à la section précédente. Il existe deux moyens d'initialiser un objet de type **Isotrope**, soit par l'intermédiaire du constructeur (avec arguments) ou soit en utilisant la fonction suivante :

Modifier_Par_Phys (...) Cette fonction permet d'initialiser, si ce n'est pas déjà fait, ou de modifier les paramètres physiques de l'objet de type **Isotrope**. Ainsi, cette fonction prend comme arguments le module d'Young (**Etmp**), le coefficient de Poisson (**nutmp**) et le nom (**name**) du matériau.

B.3 Classe Orthotrope

```

class Orthotrope : public Materiauxelastique
{
private:
    String    nom;
    char      *Type;
    Boolean    Hcalcule;    // Vrai si la matrice [H] est calcule
    Boolean    Htcalcule;  //matrice [Ht] est calcule
    real      E_L;        // Module E longitudinal
    real      E_T;        // Module E transversal
    real      nu_LT;      // Coefficient de poisson nu LT
    real      nu_TZ;      // Coefficient de poisson nu TZ

```



```

real    nu_TL;        // Coefficient de poisson nu TL
real    G_LT;        // Module de cisaillement G LT
real    G_TZ;        // Module de cisaillement G TZ
Mat(real) mat_HL;    // Matrice [H]
Mat(real) mat_HtL;   // Matrice [H_tau_L]

public:
  Orthotrope(real EL, real ET, real nuLT, real nuTZ,
             real GLT, String name);
  Orthotrope();
  ~Orthotrope() {};

  Modifier_Par_Phys (real EL, real ET, real nuLT, real nuTZ,
                    real GLT, String name);

  Calcul_H(Mat(real)& Mat_HL);
  Calcul_Ht(Mat(real)& Mat_HtL);

  String Getmateriau() { return nom;}
  print();
};

```

La classe *Orthotrope*, très semblable à la classe *Isotrope*, permet de définir un matériau orthotrope dans le plan TL et isotrope dans le plan Tz . Ce type de matériau est très largement utilisé dans la définition d'un matériau composite constitué d'un empilement de ce type de matériau (voir section A.2.2). La seule différence entre la classe *Isotrope* et *Orthotrope* est le nombre de paramètres passés en arguments.

En effet, pour définir un matériau de type orthotrope, nous devons définir le module d'Young longitudinal (EL), le module d'Young transversal (ET), les coefficients de Poisson ν_{LT} et ν_{Tz} ($nuLT$ et $nuTz$), le module de cisaillement G_{LT} (GLT) et le nom ($name$).

B.4 Classe Pli

```

class Pli : public Materiauxelastique
{

```

```

private:
    String      nom;          // Nom du pli
    Boolean     Hcalcule;    // Vrai si la matrice [H] est calcule
    Boolean     Htcalcule;   // Vrai si la matrice [Ht] est calcule
    Boolean     Acalcule;    // Vrai si la matrice [A] est calcule
    real       theta;        // Angle du pli en degre
    real       ratioepaisseur; // ratio d'epaisseur du pli
    Mat(real)  mat_H;        // Matrice [H]
    Mat(real)  mat_Ht;       // Matrice [Ht]
    Mat(real)  mat_A;        // Matrice [A] (pour le calcule de [Hc])
    Mat(real)  T1;          // Matrice de rotation [T1]
    Mat(real)  T2;          // Matrice de rotation [T2]
    Materiauxelastique *Ptrmateriau; // Pointeur sur le materiau
                                           // du pli

    real       Deg2rad(real angle); // Fonction de conversion
                                           // degre -> radian

public:
    Pli(Materiauxelastique& Ptrmat, real angle, real rt, String name);
    Pli(Materiauxelastique& Ptrmat, real rt, String name);
    Pli();
    ~Pli() {};

    Modifier_Par_Phys (Materiauxelastique& Ptrmat, real angle,
                      real rt, String name);
    Modifier_Par_Phys (Materiauxelastique& Ptrmat, real rt, String name);

    Calcul_H(Mat(real)& Mat_H);
    Calcul_Ht(Mat(real)& Mat_Ht);

    real Getratioepaisseur() { return ratioepaisseur; }
    Calcul_sigL(Vec(real) vec_sig, Vec(real)& Vec_sigL);
    Calcul_tauL(Vec(real) vec_tau, Vec(real)& Vec_tauL);
    Calcul_A(Mat(real) Mat_Hf);
    Calcul_Clib(Mat(real)& Mat_Cib);
    String Getmateriau() { return nom; }
    print();
};

```

Cette classe permet de créer un pli qui pourra directement être utilisée pour définir un laminé. Un pli est caractérisé par un matériau (ici **Materiauxelastique**), un angle et un épaisseur. Ainsi, pour définir un objet de type **Pli**, il y a deux choix possibles. Si le matériau constituant le pli est de type **Isotrope**, il faut passer en argument un pointeur sur un matériau de type **Isotrope** (**Ptrmat**), un ratio d'épaisseur (**rt**) et un nom (**name**). Par contre, si le matériau est de type **Orthotrope**, il faut passer en argument un pointeur sur un matériau de type **Orthotrope** (**Ptrmat**), un angle (**angle**), un ratio d'épaisseur (**rt**) et un nom (**name**). Ici, le ratio d'épaisseur désigne, comme son nom l'indique, le ration entre l'épaisseur du pli et l'épaisseur totale du laminé. Ici encore, l'initialisation de l'objet se fait soit par l'intermédiaire du constructeur ou de la fonction **Modifier.Par.Phys**.

Comme nous pouvons le constater, d'autres fonctions publiques peuvent être appelées. La plupart de ces fonctions sont utilisées par la classe **Composite** qui sera décrite plus loin. Voici une brève description de chacune d'elles :

real Getratioepaisseur() Cette fonction permet d'obtenir le ratio d'épaisseur d'un pli.

Calcul_sigL(...) Cette fonction calcule les contraintes des fibres à partir des contraintes calculés dans le repère global du composite (voir section A.2.3). Les contraintes sont obtenues par l'intermédiaire du vecteur **Vec_sigL**.

Calcul_tauL(...) Cette fonction calcule les contraintes de cisaillement dans la direction des fibres à partir des contraintes de cisaillement calculer dans le repère global du composite (voir section A.2.3). Les contraintes sont obtenues par l'intermédiaire du vecteur **Vec_tauL**.

Calcul_A(...) Cette fonction calcule la matrice $[A(z)]$ à partir de l'équation A.56 pour le calcul des contraintes dans l'épaisseur d'un composite.

Calcul_C1ib(...) Cette fonction calcule la matrice $[\tilde{C}_{1i}]$ de l'équation A.71 servant au calcul de la matrice $[H_c]$.

B.5 Classe Composite

```
class Composite : public Materiaux
```

```

{
protected:
    Boolean EpaisseurConst; // Vrai si l'épaisseur est constante
    Boolean Hmcalcule;      // Vrai si la matrice [Hm] calcule
    Boolean Hfcalcule;      // Vrai si la matrice [Hf] calcule
    Boolean Hmfcalcule;     // Vrai si la matrice [Hmf] calcule
    Boolean Hcbcalcule;     // Vrai si la matrice [Hcb] calcule
    Boolean Hccalcule;      // Vrai si la matrice [Hc] calcule
    String nom;             // Nom du composite
    real    epaisseur;      // Epaisseur du composite
    int     Nbpli;          // Nombre de plis
    dllist<Pli *> List_plis; // Liste chaine constitue de pointeur
                                // sur des plis

    Mat(real) mat_Hm;       // Matrice [Hm]
    Mat(real) mat_Hf;       // Matrice [Hf]
    Mat(real) mat_Hmf;      // Matrice [Hmf]
    Mat(real) mat_Hcb;      // Matrice [Hc] sans facteur k
    Mat(real) mat_Hc;       // Matrice [Hc] avec facteur k

    // Calcul de la matrice [M] pour le calcul de [Hc]
    Calcul_M(real z, Mat(real)& Mat_M);

public:
    Composite(dllist<Pli *> list_plis, real t, String name,
              Boolean epaiconst = dpFALSE);

    Composite();
    Setepaisseur(real t) { epaisseur = t; }
    Calcul_Hm(Mat(real)& Mat_Hm);
    Calcul_Hf(Mat(real)& Mat_Hf);
    Calcul_Hmf(Mat(real)& Mat_Hmf);
    Calcul_Hcb(Mat(real)& Mat_Hcb);
    Calcul_Hc(Mat(real)& Mat_Hc);
    real Calcul_k(int i);

    Calcul_sig_z(real z, Vec(real) vec_e, Vec(real) vec_x,
                 Vec(real)& Vec_sig);
    Calcul_tau_z(real z, Vec(real) gamma, Vec(real)& Vec_tau);
    Calcul_sigL_z(real z, Vec(real) vec_e, Vec(real) vec_x,
                  Vec(real)& Vec_sigL);
    Calcul_tauL_z(real z, Vec(real) gamma, Vec(real)& Vec_tauL);

```

```

Calcul_effort_N(Vec(real) vec_e, Vec(real) vec_x,
                Vec(real)& Vec_N);
Calcul_effort_M(Vec(real) vec_e, Vec(real) vec_x,
                Vec(real)& Vec_M);
Calcul_effort_T(Vec(real) gamma, Vec(real)& Vec_T);

String Getmateriau() { return nom;}
print();
};

```

La classe `Composite` permet de créer un matériau composite (laminé) formé d'un empilement de plis orthotropes ou isotropes. La seule façon d'initialiser un objet de type `Composite` est en utilisant le constructeur. Le constructeur prend comme argument un objet de type `dlist<Pli *>` qui est une liste doublement chaînée de pointeur sur des objets de type `Pli`. Cette liste contient les pointeurs sur les plis constituant le laminé du bas vers le haut. Les autres paramètres passés en argument sont l'épaisseur totale du composite, le nom du laminé et un paramètre facultatif qui indique si le laminé à une épaisseur constante ou non.

Cette classe contient beaucoup de fonctions qui servent à calculer différents paramètres et état de contraintes au cours d'une analyse. En voici une brève description :

`Calcul_M(...)` Cette fonction membre privée sert pour le calcul de la matrice $[H_c]$ et plus particulièrement pour le calcul de l'équation A.72.

`Setepaisseur(real t)` Cette fonction est utile seulement lorsque le laminé n'a pas une épaisseur constante. Elle permet donc de changer l'épaisseur selon l'endroit.

`Calcul_Hm(...)` Cette fonction calcule la matrice $[H_m]$ à l'aide de l'équation A.45.

`Calcul_Hf(...)` Cette fonction calcule la matrice $[H_f]$ à l'aide de l'équation A.47.

`Calcul_Hmf(...)` Cette fonction calcule la matrice $[H_{mf}]$ à l'aide de l'équation A.46.

`Calcul_Hcb(...)` Cette fonction calcule la matrice $[H_c]$ sans les facteurs de correction de cisaillement k .

- Calcul_Hc(...)** Cette fonction calcule la matrice $[H_c]$ en tenant compte des facteurs de correction de cisaillement k_i selon la méthode présentée à la section A.3.1.
- real Calcul_k(...)** Cette fonction calcule les facteurs de correction de cisaillement k_1 , k_2 et k_{12} . L'indice i ne prend donc que les valeurs 1, 2 ou 12 (voir équation A.68 page 108).
- Calcul_sig_z(...)** Cette fonction calcule les contraintes σ_x , σ_y et τ_{xy} a une hauteur déterminée. Elle prend comme arguments le vecteur des déformation de membranes $\{e\}$ et le vecteur des courbures $\{\kappa\}$ (voir équation A.55 page 106).
- Calcul_tau_z(...)** Cette fonction calcul les contraintes τ_{xz} et τ_{yz} a une hauteur déterminée. Elle prend comme arguments le vecteur des déformation de cisaillement transverse $\{\gamma\}$ (voir équation A.57 page 106).
- Calcul_sigL_z(...)** Cette fonction calcule le vecteur des contraintes $\{\sigma_L\}$ dans la direction des fibres a une hauteur déterminée. Elle prend comme arguments le vecteur des déformation de membranes $\{e\}$ et le vecteur des courbures $\{\kappa\}$ (voir équation A.37 page 102).
- Calcul_tauL_z(...)** Cette fonction calcule le vecteur des contraintes $\{\tau_L\}$ dans la direction des fibres a une hauteur déterminée. Elle prend comme arguments le vecteur des déformation de cisaillement transverse $\{\gamma\}$ (voir équation A.38 page 102).
- Calcul_effort_N(...)** Cette fonction calcule le vecteur des efforts normaux $\{N\}$ à partir de l'équation A.39.
- Calcul_effort_M(...)** Cette fonction calcule le vecteur des moments fléchissant $\{M\}$ à partir de l'équation A.40.
- Calcul_effort_T(...)** Cette fonction calcule le vecteur des efforts tranchants $\{T\}$ à partir de l'équation A.41.