

République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



Université A. Mira de Béjaïa



Faculté de Technologie

Département de Génie Mécanique

Mémoire de fin d'études présenté en vue de l'obtention du diplôme de Master

Spécialité : IMAO (Ingénierie Mécanique Assistée par Ordinateur)

Thème :

**Programmation de la Méthode des
Éléments Finis sous MATLAB**

Présenté par :

Mr ZEROUROU Bachir

M^{elle} OUDJEDI Lydia

Encadré par :

Mr S. SFARNI

Jury composé de:

- **Mr HAMRI Okba, Université A. Mira de Béjaïa**
- **Mr BECHEUR Abdelhamid, Université A. Mira de Béjaïa**
- **Mr BENSLIMANE Abdelhakim, Université A. Mira de Béjaïa**

Année universitaire 2015/2016



Remerciement

Avant de présenter notre projet de fin d'étude, nous tenons, à exprimer notre très grande gratitude à notre promoteur Mr S. Sfarni pour les encouragements et conseils, toute au long de notre élaboration de ce modeste manuscrit. Nous remercions et nous saluons aussi Mr Seghir pour sa disponibilité et son aide.

Nous remercions vivement, les membres de jury Mr Bécheur, Mr Hamri ainsi que Mr Benslimane qui ont bien voulu nos honorer en examinant notre travail.

Nous tenons aussi à remercier nos deux familles pour leur soutien moral et matériel sans égal durant tous nos études et notre vie.

Nous remercions tous nos ami(e)s et camarades qui nous avons aidé.



Dédicaces

Avec une grande joie je dédie ce modeste travail :

- *A mes parents*
- *A mes frères et mes sœurs*
- *A toute ma grande famille*
- *A mon binôme Lydía*
- *A mes amis des chambres J314,C301 et H14*
- *A tous mes ami(e)s*





Dédicaces

C'est avec profonde gratitude et sincères mots, que je dédie ce modeste travail :

- Mes chers parents ; qui ont sacrifié leur vie pour mon réussite*
- Mes frères ; Massinissa, Oualí et Samy pour leur grand amour et leur soutiens*
 - Toute ma famille.*
 - Mon cher Smaíl.*
 - Mon binôme Bachír*
- Mes amis ; Massinissa, Thanina et Lynda.*
 - Mes copines de chambre.*



Table des matières

Liste des figures

Liste des tableaux

Notations

Introduction générale.....1

Chapitre I Principe de la méthode des éléments finis

I Introduction.....	3
I.1 Equations différentielles et problèmes aux limites	3
I.1.1 Modélisation d'un système physique.....	3
I.1.2 Equation de ce système physique.....	3
I.1.3 Conditions aux limites.....	4
I.2 Méthode des travaux virtuels.....	4
I.3 Etapes logiques du calcul par éléments finis.....	6
I.4 Formulation intégrale	7
I.4.1 Forme forte.....	7
I.4.2 Forme faible	8
I.5 Discrétisation du domaine	9
I.5.1 Procéder de discrétisation.....	9
I.5.2 Types d'éléments classiques.....	10
I.5.3 Règles de discrétisation.....	11
I.5.3 Erreur de discrétisation.....	11
I.6 Approximation des fonctions	11
I.6.1 Approximation polynomiale.....	12
I.6.2 Approximation nodale.....	13
I.7 Elément de référence.....	13
I.7.1 Définition.....	13
I.7.2 Elément isoparamétrique.....	15
I.7.3 Transformation des opérateurs de dérivation et d'intégration	16
I.7.4 Construction des matrices élémentaires	16
I.8 Assemblage et résolution.....	17

Table des matières

I.9 Convergence de la méthode des éléments finis.....	18
Conclusion.....	18

Chapitre II Rappels sur la mécanique des solides

II Introduction.....	19
II.1 Les contraintes	19
II.1.1 Notion de contrainte.....	19
II.1.2 Bilan d'équations internes	20
II.1.3 Bilan d'équations externes	21
II.2 Les déformations.....	22
II.2.1 Déplacements.....	22
II.2.2 Relations entre déplacements et déformations.....	23
II.3 Loi de comportement.....	24
II.3.1 Relations entre déformations et contraintes.....	25
II.3.2 Relations entre contraintes et déformations.....	25
II.3.3 Etat plan de contraintes et de déformations.....	26
II.4 Energie de déformation élastique.....	27
II.4.1 Cas du chargement uniaxial.....	27
II.4.2 Cas général.....	28
Conclusion.....	28

Chapitre III Formulation des éléments barre, poutre et membrane par la MEF

III Introduction.....	29
III.1 Eléments de barre à deux nœuds	29
III.1.1 Les étapes d'analyse par éléments finis d'une barre	29
III.1.1.1 Equations gouvernantes	29
III.1.1.2 Formulation de l'élément	30
III.1.2 Treillis planes.....	31
III.1.3 Organigramme de résolution de l'élément barre	34
III.2 Eléments de poutre à deux nœuds	35

Table des matières

III.2.1 Définition.....	35
III.2.2 Les étapes d'analyse d'une poutre à deux nœuds	35
III.2.2.1 Equations gouvernantes.....	35
III.2.2.2 Formulation intégrale.....	37
III.2.2.3 Discrétisation.....	38
III.2.2.4 Matrices élémentaires.....	39
III.2.3 Organigramme de résolution de l'élément poutre	42
III.3 Elément membrane.....	43
III.3.1 Définition.....	43
III.3.2 Etapes d'analyse de l'élément membrane.....	43
III.3.2.1 Equation gouvernante.....	43
III.3.2.2 Fonctions de forme.....	44
III.3.2.3 Matrice de rigidité et vecteur charge.....	44
III.3.3 Analyse avec l'élément quadrangle $Q4$	45
III.3.3.1 Fonctions de forme.....	45
III.3.3.2 Matrice Jacobienne et son invers.....	45
III.3.3.3 Matrice de rigidité élémentaire	46
III.3.3.4 Vecteur force élémentaire	47
III.3.4 Organigramme de résolution.....	48

Chapitre IV Présentation des programmes de calcul développés

IV Introduction.....	49
IV.1 Etapes caractéristiques d'un programme d'éléments finis.....	49
IV.2 Techniques de programmation.....	50
IV.2.1 Possibilité du programme.....	50
IV.2.2 Modularité	50
IV.2.3 Langage de programmation.....	51
IV.3 Programmes développés.....	51
IV.3.1 Programme d'élément barre à deux nœuds.....	50
IV.3.1.1 Organisation.....	51
IV.3.1.2 Fichier définissant le problème à résoudre.....	52
IV.3.1.3 Les algorithmes des différentes fonctions.....	53

Table des matières

IV.3.2 Programme d'élément poutre à deux nœuds.....	55
IV.3.2.1 Organisation.....	55
IV.3.2.2 Fichier définissant le problème à résoudre	56
IV.3.2.3 Les algorithmes des différentes fonctions.....	57
IV.3.3 Programme d'élément membrane.....	60
IV.3.3.1 Organisation.....	60
IV.3.3.2 Fichier définissant le problème à résoudre.....	60
IV.3.3.3 Les algorithmes des différentes fonctions.....	61
Chapitre V : Applications	
V.1 Application : élément barre.....	65
V.1.1 Exemple 1 (treillis a sept barres)	65
V.1.2 Exemple 2 (treillis à onze barres)	66
V.1.3 Interprétation des résultats	68
V.2 Application : élément poutre	68
V.2.1 Exemple 1(poutre encastree)	68
V.2.2 Exemple 2 (portique)	69
V.2.3 Interprétation des résultats	71
V.3 Application : élément membrane	71
V.3.1 Exemple 1 (traction d'une plaque carrée).....	71
V.3.1.1 Interprétation des résultats	75
V.3.2 Exemple 2 (traction d'une plaque trouée).....	75
V.3.2.1 Interprétation des résultats	78
Conclusion générale	79
Bibliographie	

Liste des figures

N°	Figure	Page
Chapitre I		
I.1	Charges et conditions aux limites.	4
I.2	Les étapes logiques du calcul par éléments finis.	7
I.3	Equations gouvernantes sur un domaine et son contour.	8
I.4	Discrétisation du domaine.	9
I.5	Règles de discrétisation.	11
I.6	Méthode de réduction d'erreurs de discrétisation.	11
I.7	Transformation géométrique.	14
Chapitre II		
II.1	Équilibre d'un solide.	20
II.2	Vecteur contrainte.	20
II.3	Equilibre de volume.	21
II.4	Equilibre de surface.	22
II.5	Déformations planes.	23
II.6	Energie de déformation uniaxial.	28
Chapitre III		
III.1	Equilibre élémentaire d'une barre.	29
III.2	Systèmes à treillis.	32
III.3	Elément barre bidimensionnelle.	32
III.4	Elément barre incliné.	33
III.5	Poutre chargée.	35
III.6	Déformation d'une poutre.	36
III.7	Equilibre statique.	37
III.8	Elément poutre à deux nœuds.	38
III.9	Elément membrane.	43
III.10	Elément quadrangle Q4.	45
III.11	Points de Gauss-Legendre.	47
Chapitre IV		
IV.1	Etapas caractéristiques d'un programme d'éléments finis	49

Chapitre V		
V.1	Modélisation du treillis à sept barres avec ANSYS	63
V.2	Déplacements (mm) de la structure à sept barres	64
V.3	Modélisation du treillis à onze barres avec ANSYS	65
V.4	Déplacements (mm) de la structure à onze barres.	66
V.5	Modélisation de la poutre encastree avec ANSYS	66
V.6	Déplacements et rotations (mm) de la poutre encastree.	67
V.7	Modélisation du portique avec ANSYS	68
V.8	Déplacements et rotations (mm) du portique.	69
V.9	Traction de la plaque carrée	70
V.10	Déplacements (mm) et contraintes (Mpa) de la plaque pour un maillage (1x1), ($\nu = 0$)	70
V.11	Déplacements (mm) et contraintes (MPa) de la plaque pour un maillage (2x2), ($\nu = 0$)	71
V.12	Déplacements (mm) et contraintes (MPa) de la plaque pour un maillage (10x10), ($\nu = 0$)	71
V.13	Déplacements (mm) et contraintes (MPa) de la plaque pour un maillage (1x1), ($\nu = 0.3$)	72
V.14	Déplacements (mm) et contraintes (MPa) de la plaque pour un maillage (2x2), ($\nu = 0.3$)	72
V.15	Déplacements (mm) et contraintes (MPa) de la plaque pour un maillage (10x10), ($\nu = 0.3$)	72
V.16	Plaque rectangulaire trouée	73
V.17	Maillage d'un cercle avec l'élément Q4 pour un angle de 45°	74
V.18	Schéma des maillages, (a) maillage 8x4, (b) maillage 12x4, (c) maillage 44x15	76
V.19	L'orientation des maillages utilisés.	77
V.20	Déplacements (mm) et contraintes (Mpa) de la plaque trouée pour un maillage (8x4), ($\nu = 0.3$)	78
V.21	Déplacements (mm) et contraintes (Mpa) de la plaque trouée pour un maillage (12x4), ($\nu = 0.3$)	78
V.22	Déplacements (mm) et contraintes (MPa) de la plaque trouée pour un maillage (44x15), ($\nu = 0.3$)	78

Liste des tableaux

N°	Tableau	Page
Chapitre I		
Tableau I.1	Table des coordonnées des nœuds géométriques.	9
Tableau I.2	Table des connectivités.	9
Tableau I.3	Tableau de quelques éléments classique.	10
Tableau I.4	Exemple de types d'éléments de référence.	15
Chapitre IV		
Tableau IV.1	L'ensemble des fichiers du programme MEFbarre.	51
Tableau IV.2	L'ensemble des fichiers du programme MEFpoutre.	54
Tableau IV.3	L'ensemble des fichiers du programme MEFmembrane.	59
Chapitre V		
Tableau V.1	Liste des éléments utilisés dans ANSYS	64
Tableau V.2	Résultats des déplacements (mm) et les réactions (N) du treillis à sept barres.	66
Tableau V.3	Résultats des déplacements (mm) et les réactions (N) du treillis à onze barres.	67
Tableau V.4	Résultats des déplacements et les rotations (mm) de la poutre encastree.	69
Tableau V.5	Résultats des déplacements et les rotations (mm) du portique.	70
Tableau V.6	Résultat du déplacement (mm) et contrainte (MPa) maximal de la plaque pour $\nu = 0$.	72
Tableau V.7	Résultat du déplacement (mm) et contrainte (MPa) maximal de la plaque pour $\nu = 0.3$	73
Tableau V.8	Résultat du déplacement (mm) et contrainte (MPa) maximal de la plaque trouée pour $\nu = 0.3$	76

Nomenclature

MEF : Méthode des Eléments Finis.

RDM : Résistance Des Matériaux.

PTV : Principe des Travaux Virtuels.

CAO : Conception Assistée par Ordinateur.

UML : Unified Modeling Language.

EDI : Environnement de développement intégré.

Ω, Γ : Domaine et frontière d'un corps solide.

F_v : Force volumique [N].

F_s : Force surfacique [N].

W_{int}, W_{ext} : Travail des forces internes et externes [W].

$\{u\}$: Vecteur colonne.

$\langle u \rangle$: Vecteur ligne.

[Der] : Operateur de dérivabilité.

[P(x)] : Matrice des fonctions polynomiales.

$\{a\}$: Paramètres généraux de l'approximation.

[N] : Fonction de forme.

[B] : Matrice de dérivabilité des fonctions de formes.

ξ, η : Coordonnées paramétriques.

[J] : Matrice Jacobienne.

[H] : Matrice du comportement élastique.

[K] : Matrice de rigidité.

$\{U\}$: Déplacements.

$\{F\}$: Vecteur du chargement.

[K_e] : Matrice de rigidité élémentaire.

$\{F_e\}$: Le vecteur de chargement élémentaire.

$\{\sigma\}$: Vecteur des contraintes.

$\{\varepsilon\}$: Vecteur des déformations.

τ : Contrainte de cisaillement $[\text{N}/\text{mm}^2]$.

E : Module d'élasticité longitudinale $[\text{GPa}]$.

G : Module d'élasticité transversale $[\text{GPa}]$.

ν : Coefficient de Poisson.

W : Energie de déformation élastique $[\text{J}]$.

x, y, z : Coordonnées cartésiennes $[\text{mm}]$.

u, v, θ_1, θ_2 : Déplacements et rotation d'un nœud $[\text{mm}]$.

L : Longueur $[\text{mm}]$.

A : Section $[\text{mm}^2]$.

h : Hauteur $[\text{mm}]$.

b : Largeur $[\text{mm}]$.

t : Epaisseur $[\text{mm}]$.

ρ : Masse volumique $[\text{kg}/\text{m}^3]$.

I : Moment d'inertie $[\text{mm}^4]$.

$[R]$: Matrice de rotation.

M : Moment de flexion $[\text{N}\cdot\text{mm}]$.

T : Effort tranchant $[\text{N}]$.

w_i, w_j : Poids des points de Gauss-Legendre.

Introduction générale

Introduction générale

L'ingénieur de bureau d'étude doit concevoir puis dimensionner des systèmes mécaniques, pour bien mener ses projets, très souvent celui-ci utilisera des outils de calculs basés sur la méthode des éléments finis dont l'utilisation s'est généralisée dans l'industrie depuis une vingtaine d'années.

Cette méthode lui permettra de résoudre un éventail très large de problèmes : structurels, thermiques, électromagnétiques, fluidiques, avec des aspects linéaires ou non linéaire, stationnaires ou transitoires.

Différents éditeurs de logiciels proposent généralement plusieurs modules sur le marché, permettant de traiter des problèmes multi-physiques.

La structure de ces codes comporte habituellement un pré-processeur, un ou plusieurs solveurs, un ou plusieurs post-processeurs. Le pré-processeur est une interface graphique permettant à l'utilisateur de décrire (modéliser) la géométrie. Les solveurs intègrent les bases des méthodes de résolution (linéaire, non linéaire, stationnaire ou transitoire, etc.) spécifiques au cas étudié. Les post-préprocesseurs permettent de visualiser les résultats sous forme de courbes ou de cartographies colorées.

Mais avant d'utiliser un tel code de calcul de manière opérationnelle, il est essentiel d'explorer ses capacités et surtout ses limites. Pour ce faire l'utilisateur devra maîtriser un minimum d'un prérequis théorique dans le secteur visé (génie-mécanique, génie-civil, etc.) mais également dans le domaine de la méthode des éléments finis. Toujours dans ce même domaine et au niveau pratique, il devra être capable de résoudre des problèmes simples avec le logiciel mis à sa disposition.

Introduction générale

Le présent travail fait l'objet d'une formulation des éléments barre, poutres et membranes par la MEF.

Trois programmes de calcul par élément finis, ont été développés pour le calcul de la réponse statique de chaque élément en contrainte et déplacement.

Le présent manuscrit est organisé en cinq chapitres : Au premier chapitre, nous présentons le principe de la méthode des éléments finis, le deuxième chapitre contient des rappels sur la mécanique des solides. Dans le troisième chapitre la formulation des différents éléments barre, poutre et membrane est développée. Le quatrième chapitre est consacré à la présentation des programmes développés ainsi que leur utilisation. Le dernier chapitre comporte la partie pratique de notre travail, dans laquelle nous appliquons les programmes, on outre les résultats obtenus sont comparés à ceux de ANSYS. Finalement vient la clôture de notre manuscrit avec une conclusion générale.

Chapitre I: Principe de la méthode des éléments finis

Chapitre I : Principe de la méthode des éléments finis

I Introduction

Dans ce premier chapitre nous présentons le principe de la méthode des éléments finis, qui est une technique numérique qui permet de résoudre des équations différentielles, elle consiste à subdiviser (discrétiser) le domaine de ces équations en plusieurs sous-domaines appelés éléments finis qui sont interconnectés entre eux par des nœuds, la solution recherchée est remplacée dans chaque élément par une approximation avec des polynômes simples et le domaine peut ensuite être reconstitué avec l'assemblage de tous ces éléments. La MEF appliquée aux calculs de structures, est une technique multidisciplinaire, elle fait appel aux trois disciplines suivantes :

- La mécanique des structures : élasticité linéaire, RDM (science de l'ingénieur).
- L'analyse numérique : méthode d'approximation, intégration numérique, résolution des systèmes linéaires,...etc.
- Science informatique : techniques de développement (génie-logiciels) et de maintenance.

I.1 Equations différentielles et problèmes aux limites

I.1.1 Modélisation d'un système physique

Un système physique peut être modélisé par des équations différentielles ou des équations aux dérivées partielles qui agissent sur des inconnues u qui peuvent être [1] :

- Un scalaire comme la température ou la pression.
- Un vecteur comme les déplacements et les vitesses.

I.1.2 Equations de ce système physique

Une équation différentielle est une équation impliquant une ou plusieurs dérivées d'une fonction inconnue, une équation mettant en jeu des dérivées partielles est appelée équation aux dérivées partielles [2].

Un système physique continu est le plus souvent écrit par un système d'équations aux dérivées partielles.

Chapitre I : Principe de la méthode des éléments finis

I.1.3 Conditions aux limites

Pour en retenir une solution de ces équations, nous devons imposer des conditions qui correspondent à la valeur prise par la solution en un point sur intervalle d'intégration.

Pour illustrer ça considérons un domaine Ω de frontière Γ (Figure I.1a) soumis à :

- un champ de forces volumiques : F_V (poids propre).
- des déplacements imposés sur la frontière : U
- des forces surfaciques sur la frontière : F_S

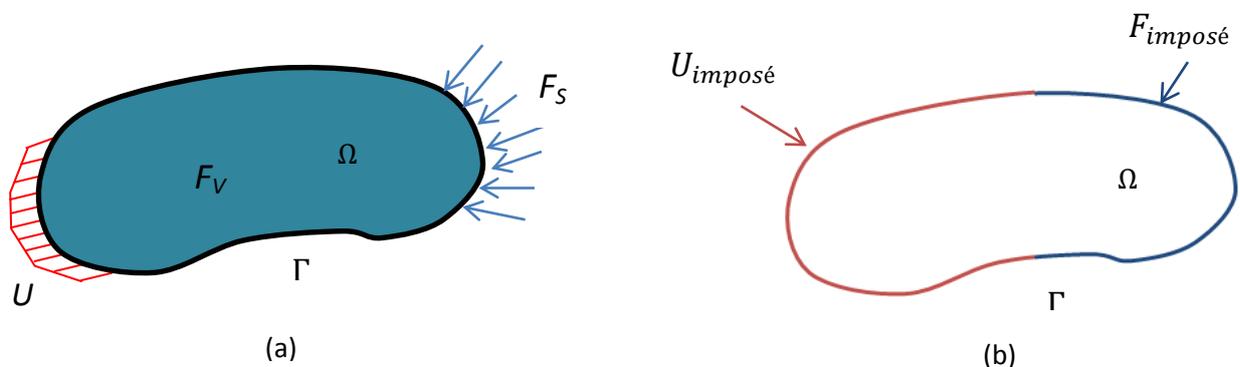


Figure I.1 : Charges et conditions aux limites.

Ainsi la surface de la frontière Γ ne peut admettre que deux types de conditions aux limites voir (figure I.1b) :

- Conditions aux limites portant sur les déplacements $U_{imposé}$ (dites cinématiques).
- Conditions aux limites portant sur les efforts $F_{imposé}$ (dites statiques).

I.2 Méthode des travaux virtuels

Le principe des travaux virtuels (PTV) est le point de départ de la méthode des éléments finis, l'expression du PTV traduit l'équilibre du solide soumis à l'action des forces de volume et de surface sous une forme intégrale [3].

$$W_{int} = W_{ext} \quad (I.1)$$

où W_{int} représente le travail des efforts internes et W_{ext} le travail des forces externes.

Chapitre I : Principe de la méthode des éléments finis

ce qui se traduit par :

$$\int_{\Omega} \{\varepsilon\}^T \{\sigma\} dV = \int_{\Omega} \{u\}^T F_V dV + \int_{\Gamma} \{u\}^T F_S dS \quad (I.2)$$

Le principe des Travaux Virtuels consiste à imposer un déplacement cinématiquement admissible (continu et dérivable) $\{u^*\}$ sur la structure et à évaluer les travaux des efforts internes et externes pour trouver la condition d'équilibre. Ce déplacement est appelé « déplacement virtuel »; il va provoquer une déformation virtuelle $\{\varepsilon^*\}$. L'équation (I.2) devient :

$$\int_{\Omega} \{\varepsilon^*\}^T \{\sigma\} dV = \int_{\Omega} \{u^*\}^T F_V dV + \int_{\Gamma} \{u^*\}^T F_S dS \quad (I.3)$$

avec l'approximation nodale que nous allons définir à la section I.6 :

$$\{u\} = [N]\{U\} \quad (I.22)$$

le déplacement virtuel s'écrit :

$$\{u^*\} = [N]\{U^*\} \quad (I.4)$$

$$\text{la relation liant les déplacements et les déformations : } \{\varepsilon\} = [Der]\{u\} \quad (I.5)$$

où $[Der]$ est la matrice de dérivabilité.

devient :

$$\{\varepsilon^*\} = [Der]\{u^*\} \quad (I.6)$$

$$\text{pour la relation contraintes déformation : } \{\sigma\} = [H]\{\varepsilon\} \quad (I.7)$$

en remplaçant l'équation (I.22) dans (I.5) puis tout dans (I.7) nous obtiendrons :

$$\{\sigma\} = [H][Der][N]\{U\} = [H][B]\{U\} \quad (I.8)$$

Finalemment l'équation (I.3) devient :

$$\int_{\Omega} ([Der][N]\{U^*\})^T [H][Der][N]\{U\} dV = \int_{\Omega} ([N]\{U^*\})^T F_V dV + \int_{\Gamma} ([N]\{U^*\})^T F_S dS$$

après simplification de $\{U^*\}$:

$$\int_{\Omega} [B]^T [H][B]\{U\} dV = \int_{\Omega} [N]^T F_V dV + \int_{\Gamma} [N]^T F_S dS \quad (I.9)$$

ce qui mène à un système algébrique :

$$[K]\{U\} = \{F\} \quad (\text{I.10})$$

$[K]$: est la matrice caractérisant le système (matrice de rigidité) (N/m).

$\{U\}$: vecteur des déplacements inconnus (degrés de liberté) (m).

$\{F\}$: vecteur des forces appliquées (N).

si le système est en équilibre dynamique $\sum F = m \frac{d^2U}{dt^2}$ alors y aura apparition d'un nouveau

terme : $[M] = \int \rho [N]^T [N] dV$

$[M]$: matrice de masse (kg).

I.3 Etapes logiques du calcul par éléments finis [4]

- ✚ Définir le domaine (géométrie) et les équations gouvernantes : l'application de la MEF exige une réécriture sous forme intégrale, la formulation faible est la plus utilisée pour inclure les conditions aux limites.
- ✚ Définir les éléments et les nœuds (créer le maillage) : elle consiste à discrétiser le domaine, elle constitue ainsi la phase de préparation des données géométriques.
- ✚ Approximation sur un élément : nous approchons la variable inconnue par une simple fonction linéaire ou polynomiale, le degré du polynôme d'interpolation est relié au nombre de nœuds de l'élément, l'approximation nodale est appropriée, c'est dans cette étape que se fait la construction des matrices et vecteurs élémentaires.
- ✚ Assemblage et application des conditions aux limites : les propriétés de l'élément (rigidité, masse,...) sont prise en compte pour former le système algébrique, c'est à ce niveau que se fait la construction des matrices globales à partir de la table des localisations.
- ✚ Résolution du système global : pour un système linéaire, il définit soit un problème d'équilibre qui concerne un cas statique ou stationnaire, soit un problème de valeurs critiques dans lequel il faut déterminer les valeurs et vecteurs propres du système. Le problème d'équilibre statique peut être résolu par des méthodes de résolution des systèmes linéaires :
 - méthodes directes (élimination de Gauss, Cholesky,...etc).
 - méthode itératives (Gauss-Seidel, méthode de Jacobi,...etc).

Chapitre I : Principe de la méthode des éléments finis

Ces étapes sont schématisées dans la (figure I.2).

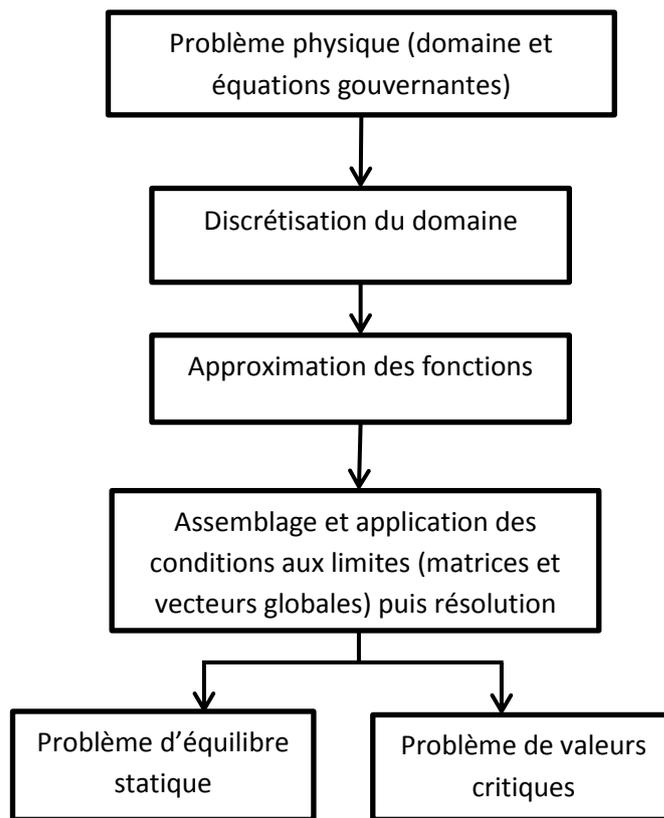


Figure I.2 : Les étapes logiques du calcul par éléments finis.

I.4 Formulation intégrale

Cette formulation est inspirée du principe des travaux virtuels qui est souvent basé sur l'égalité des travaux effectués par les forces extérieures et intérieures lors d'un déplacement virtuel. Ses déplacements virtuels sont remplacés par des fonctions arbitraires (quelconques) continue sur le domaine, l'équation est réécrite sous forme intégrale.

I.4.1 Forme intégrale forte

Un problème classique d'équations différentielles gouvernant un système physique s'énonce comme suit :

$$L(u) = 0 \text{ et } C(u) = 0 \quad (\text{I.11})$$

où $L(u)$ est l'ensemble des équations gouvernantes définies sur le domaine Ω et $C(u)$ est l'ensemble des conditions aux limites que les fonctions u doivent vérifier sur le contour Γ (figure I.3).

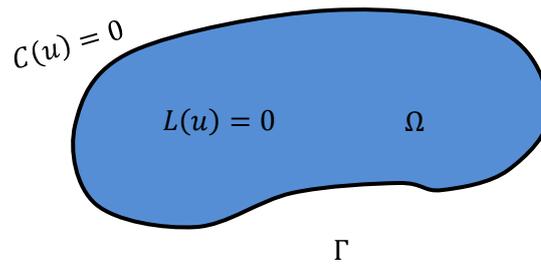


Figure I.3 Equations gouvernantes sur un domaine et son contour.

Le problème intégral associé au système (I.11) s'écrit en prenant l'intégrale du système d'équations gouvernantes par des fonctions poids, l'énoncé devient :

$$\int_{\Omega} w L(u) d\Omega = 0 \quad (\text{I.12})$$

Cette équation est appelée forme intégrale forte de l'équation différentielle (I.11). Elle est analogue à l'expression des travaux virtuels, en fait la solution de (I.12) a encore plus de portée, nous pouvons affirmer que si elle est satisfaite pour toute fonction de poids w , l'équation différentielle (I.12) est satisfaite en tout point du domaine Ω .

I.4.2 Forme intégrale faible

Pour satisfaire les conditions aux limites nous avons deux manières de procéder : soit par le choix de la fonction de pondérations (poids), soit vérifier :

$$\int_{\Omega} w L(u) d\Omega = \int_{\Omega} w \left[1, \frac{\partial}{\partial x}, \dots, \frac{\partial^n}{\partial x^n} \right] (u) d\Omega = 0 \quad (\text{I.13})$$

qu'il est possible d'intégrer l'équation (I.13) par parties:

$$\int_{\Omega} D(w)E(u) d\Omega + \int_{\Gamma} F(w)G(u) d\Gamma = 0 \quad (\text{I.14})$$

Les opérateurs D, E, F, G contiennent des dérivées d'ordre moins élevé, d'où un choix de fonctions d'approximation de u plus large.

Cette équation est la formulation faible de l'équation différentielle, elle forme la base de l'approximation par éléments finis.

Pour obtenir de telles formulations intégrales, nous disposons de deux techniques :

- ❖ Méthode des résidus pondérés (Galerkin).
- ❖ Détermination de fonctionnelles variationnelles.

Chapitre I : Principe de la méthode des éléments finis

La première est la plus utilisée ; elle consiste à choisir $w = \delta u$ fonction de Dirac (une variation quelconque de la fonction recherchée) et d'utiliser l'approximation nodale pour la discrétisation.

I.5 discrétisation du domaine [4]

I.5.1 Procédé de discrétisation

La méthode des éléments finis est une méthode d'approximation par sous domaines, donc avant toute application il faut subdiviser notre domaine à étudier en éléments. Chaque élément est défini géométriquement par un nombre de nœuds bien déterminé. (figure I.4).

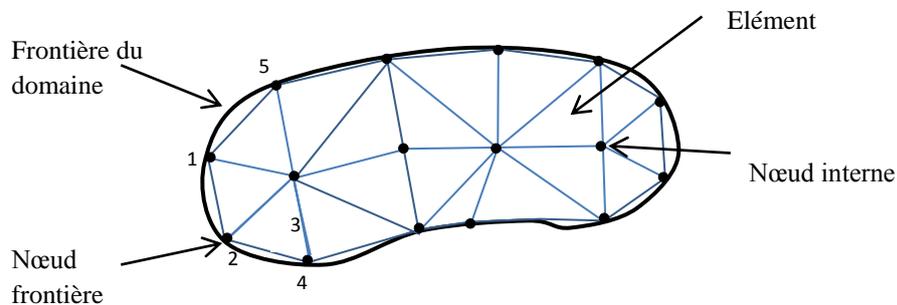


Figure I.4 Discrétisation du domaine.

Le résultat de la discrétisation doit contenir deux données essentielles :

- La table des coordonnées géométriques des nœuds.

	Nœuds				
coordonnées	1	2	3	...	n
x	x_1	x_2	x_3	...	x_n
y	y_1	y_2	y_3	...	y_n

Tableau I.1 : Table des coordonnées des nœuds géométriques.

- La table des connectivités qui stocke les numéros des nœuds qu'elle contient.

	Elt1	Elt2	Elt3	...	Elt n
Nœuds	1	2	1	.	.
	2	4	3	.	.
	3	3	5	.	.

Tableau I.2 : Table des connectivités

Chapitre I : Principe de la méthode des éléments finis

La numérotation de tous les nœuds des éléments doit être faite, il est important de choisir le sens du parcours de la liste des nœuds des éléments (trigonométrique par exemple).

I.5.2 Types d'éléments classiques

Le type d'élément à utiliser est selon la structure à discrétiser (linéique, surfacique, volumique), le (tableau I.3) résume l'ensemble des éléments classiques les plus souvent utilisés.

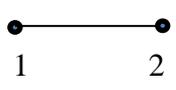
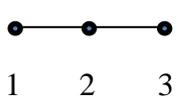
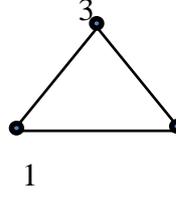
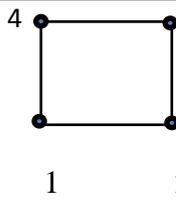
Nombres de dimensions	symbole	Degré du polynôme	Base polynomiale $\langle P \rangle$	Type du polynôme	Représentation de l'élément
1D	L2	1	$\langle 1 \ \xi \rangle$	$a_0\xi^0 + a_1\xi^1$	
	L3	2	$\langle 1 \ \xi \ \xi^2 \rangle$	$a_0\xi^0 + a_1\xi^1 + a_2\xi^2$	
2D	T3	1	$\langle 1 \ \xi \ \eta \rangle$	$a_0\xi^0 + a_1\xi^1 + a_3\eta^1$	
	Q4	2	$\langle 1 \ \xi \ \eta \ \xi\eta \rangle$	$a_0\xi^0 + a_1\xi^1 + a_3\eta^1 + a_4\xi^1\eta^1$	

Tableau I.3 : Tableau de quelques éléments classiques.

I.5.3 Règles de discrétisation [4]

La discrétisation géométrique doit respecter les règles suivantes :

-  Le nœud d'un élément ne doit pas être à l'intérieur d'un côté d'un autre élément du même type (figure I.5.a).
-  Aucun élément bidimensionnel ne doit être plat, éviter les angles proches de 0° ou de 180° (figure I.5.b).

Chapitre I : Principe de la méthode des éléments finis

- ✚ Deux éléments distincts ne peuvent avoir en commun que des points situés dans leurs frontières communes ; le recouvrement est exclu (figure I.5.c).
- ✚ L'ensemble de tous les éléments doivent constituer un domaine aussi proche que possible du domaine donné ; les vides entre éléments sont exclus (figure I.5.d).



Figure I.5 : Règles de discrétisation.

I.5.4 Erreur de discrétisation [5]

Lorsque la frontière du domaine est constituée par des courbes ou des surfaces plus complexes que celles qui définissent les frontières des éléments, une erreur est inévitable. Cette erreur est appelée «erreur de discrétisation géométrique» voir (figure I.6.a), elle peut être réduite :

- En diminuant la taille des éléments (figure I.6.b).
- En utilisant des éléments à frontières plus complexes (figure I.6.c).

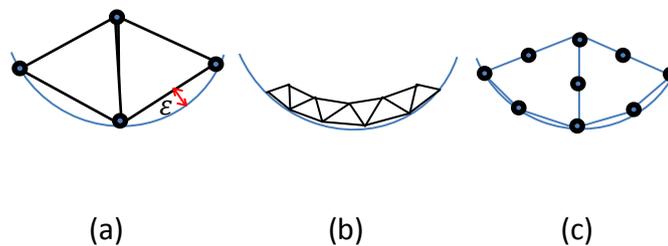


Figure I.6 : Méthode de réduction d'erreurs de discrétisation.

I.6 Approximation des fonctions

Un modèle mathématique d'un système physique fait intervenir plusieurs variables ou fonctions $u_{ex}(x)$, dite exactes : déplacements, température, vitesses, etc. Approcher une fonction exacte consiste à la remplacer par une approximative $u(x)$ telle que la différence [5] :

$$e(x) = u(x) - u_{ex}(x) \quad (\text{I.15})$$

Chapitre I : Principe de la méthode des éléments finis

soit assez petite pour l'objet visé. Il existe plusieurs types d'approximations mais leur mise en œuvre sur ordinateur est beaucoup plus difficile et coûteuse en temps de calculs, les fonctions polynomiales sont souvent les plus utilisées.

I.6.1 Approximation polynomiale

La fonction approchée est exprimée, dans le cas unidimensionnel par :

$$u(x) = a_0x^0 + a_1x^1 + \dots + a_nx^n = \sum_{i=0}^n a_i x^i \quad (\text{I.16})$$

qu'on peut écrire sous la forme vectorielle :

$$u(x) = \langle 1 \ x \ x^2 \ x^3 \ \dots \rangle \begin{Bmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{Bmatrix} = \langle P \rangle \{a\} \quad (\text{I.17})$$

dans le cas bidimensionnel :

$$u(x, y) = a_0x^0y^0 + a_1x^1y^0 + a_2x^1y^1 + \dots + a_nx^jy^k = \sum a_i x^j y^k \quad (\text{I.18})$$

sous forme vectorielle :

$$u(x, y) = \langle 1 \ x \ y \ xy \ \dots \rangle \begin{Bmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{Bmatrix} = \langle P \rangle \{a\} \quad (\text{I.19})$$

Remarque : La base polynomiale peut être choisie à l'aide du triangle de Pascal.

Ces formes d'approximation sont appelées interpolations polynomiales. Si on exprime la fonction sur tous les nœuds on obtient pour chaque nœud i de coordonnée x_i :

$$u_i = \langle P(x_i) \rangle \{a\} \quad (\text{I.20})$$

soit pour tous les nœuds :

$$\{U\} = \begin{bmatrix} \langle P_1 \rangle \\ \vdots \\ \langle P_n \rangle \end{bmatrix} \begin{Bmatrix} a_1 \\ \vdots \\ a_n \end{Bmatrix} \Rightarrow U_n = P_n \cdot a_n \quad (\text{I.21})$$

avec U_n : représente la valeur de la fonction aux nœuds.

P_n : la valeurs des polynômes aux nœuds de coordonnées x_i .

Chapitre I : Principe de la méthode des éléments finis

a_n : les variables générales qui sont les facteurs des polynômes.

L'inconvénient de cette interpolation réside dans l'utilisation des paramètres a_n comme variable de base, des calculs supplémentaires sont nécessaires pour calculer la fonction recherchée u .

I.6.2 approximation nodale

A partir de l'équation (I.21), il est facile de tirer les a_n en fonction des U_n et les remplacer dans l'équation (I.17) ce qui va donner :

$$\{u\} = \langle P \rangle [P_n]^{-1} \{U_n\} = [N] \{U_n\} \quad (I.22)$$

Ce type d'approximation est dit interpolation nodale puisqu'elle dépend des valeurs aux nœuds de la fonction inconnue u

Les fonctions N sont appelées fonctions d'interpolation (fonctions de forme), elles sont en fonction du type de l'élément utilisé pour la discrétisation géométrique.

Les éléments linéiques sont représentés directement par le polynôme de Lagrange :

$$N_i(x) = \sum_{j \neq i}^n \frac{(x - x_j)}{(x_i - x_j)} \quad (I.23)$$

I.7 Elément de référence

I.7.1 Définition [6]

Un élément de référence V^r est un élément de forme très simple, repéré dans un espace de référence, qui peut être transformé en chaque élément réel V^e par une transformation géométrique voir (figure I.7). Cette démarche nous permet de surmonter le problème des éléments à nombres de nœuds élevé ou de forme compliquée, qui conduit à des formulations très complexes aussi.

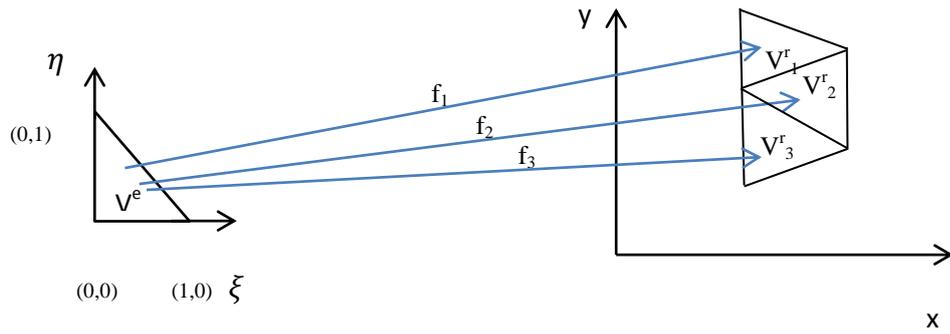


Figure I.7 : Transformation géométrique.

La fonction de transformation f définit une relation bijective entre les coordonnées (cartésienne) \mathbf{x} de chaque point de l'élément réel et les coordonnées (locales) ξ du point correspondant de l'élément de référence.

$$\mathbf{x} = f(\xi) \quad (\text{I.24})$$

avec : $\xi = \langle \xi, \eta \rangle$; $\mathbf{x} = \langle x, y \rangle$

soit :

$$\begin{cases} x = f(\xi) \\ y = f(\xi) \end{cases} = \begin{cases} x = f(\xi, \eta) \\ y = f(\xi, \eta) \end{cases} \quad (\text{I.25})$$

Les formules de transformations des coordonnées sont remplacées par des fonctions de transformations géométriques $\bar{N}(\xi)$ Pour simplifier la notation. Ce qui conduit à :

$$\mathbf{x}(\xi) = \langle \bar{N}(\xi) \rangle \{ \mathbf{x} \} \quad (\text{I.26})$$

$$\begin{cases} x = \bar{N}_1(\xi, \eta)x_i + \bar{N}_2(\xi, \eta)x_j = \langle \bar{N} \rangle \begin{Bmatrix} x_i \\ x_j \end{Bmatrix} \\ y = \bar{N}_1(\xi, \eta)y_i + \bar{N}_2(\xi, \eta)y_j = \langle \bar{N} \rangle \begin{Bmatrix} y_i \\ y_j \end{Bmatrix} \end{cases} \quad (\text{I.27})$$

Avec : $\{x_n\}$ et $\{y_n\}$: vecteurs contenant les coordonnées des nœuds.

Le (tableau I.4) illustre quelques exemples de types d'éléments de référence.

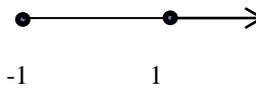
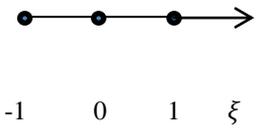
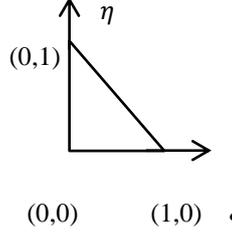
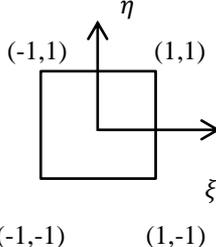
Nombres de dimensions	symbole	Représentation de l'élément
1D	L2	
	L3	
2D	T3	
	Q4	

Tableau I.4 : Exemple de types d'éléments de référence.

I.7.2 Élément isoparamétrique [6]

Un élément isoparamétrique est un élément dont les fonctions de forme N_i (d'interpolations) sont les même que les fonctions de transformations géométriques \bar{N}_i .

l'équation (I.26) peut s'écrire comme étant une somme :

$$\mathbf{x}(\xi, \eta) = \sum_{i=1}^n N_i(\xi, \eta) \mathbf{x}_i ; \mathbf{y}(\xi, \eta) = \sum_{i=1}^n N_i(\xi, \eta) \mathbf{y}_i \quad (\text{I.28})$$

Où : - n : est le nombre des nœuds.

- ξ, η : sont les coordonnées d'un point de l'élément de référence.
- $\mathbf{x}(\xi, \eta)$ et $\mathbf{y}(\xi, \eta)$: sont les coordonnées d'un point de l'élément réel.
- \mathbf{x}_i et \mathbf{y}_i : sont les coordonnées di $i^{\text{ème}}$ nœud de l'élément.
- $N_i(\xi, \eta)$: sont les fonctions de forme.

Chapitre I : Principe de la méthode des éléments finis

I.7.3 Transformation des opérateurs de dérivation et d'intégration

La matrice Jacobienne permet d'exprimer les dérivées des fonction de forme dans le repère local (coordonnées de référence), à partir de l'équation (I.27) les dérivées des fonctions de formes sont calculés comme suit :

$$\begin{cases} \frac{\partial N}{\partial \xi} = \frac{\partial N}{\partial x} \cdot \frac{\partial x}{\partial \xi} + \frac{\partial N}{\partial y} \cdot \frac{\partial y}{\partial \xi} \\ \frac{\partial N}{\partial \eta} = \frac{\partial N}{\partial x} \cdot \frac{\partial x}{\partial \eta} + \frac{\partial N}{\partial y} \cdot \frac{\partial y}{\partial \eta} \end{cases} \Rightarrow \begin{Bmatrix} N_{,\xi} \\ N_{,\eta} \end{Bmatrix} = \begin{bmatrix} x_{,\xi} & y_{,\xi} \\ x_{,\eta} & y_{,\eta} \end{bmatrix} \begin{Bmatrix} N_{,x} \\ N_{,y} \end{Bmatrix} \quad (\text{I.29})$$

Ou bien :
$$\{N_{,\xi}\} = [J]\{N_{,x}\} \quad (\text{I.30})$$

$[J]$ est la matrice Jacobienne de l'élément.

elle est obtenue en remplaçant les expressions des coordonnées des éléments réels de l'équation (I.28) :

$$[J] = \begin{Bmatrix} \frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial \eta} \end{Bmatrix} \langle x \ y \rangle = \begin{bmatrix} \langle N_{,\xi} \rangle \\ \langle N_{,\eta} \rangle \end{bmatrix} [\{x_i\} \ \{y_i\}] \quad (\text{I.31})$$

Pour exprimer les dérivées dans le plan global, nous multiplions les expressions des dérivées dans le plan de référence par l'inverse de la matrice Jacobienne.

$$\{N_{,x}\} = [J]^{-1}\{N_{,\xi}\} \quad (\text{I.32})$$

L'intégration d'une fonction de forme sur l'élément réel V^e passe à une intégration plus simple sur l'élément de référence V^r par le biais de changement de variable :

$$\int_{V^e} f(x) dx dy = \int_{V^r} f(x(\xi)) \det(J) d\xi d\eta \quad (\text{I.33})$$

I.7.4 Construction des matrices élémentaires

Dans le titre précédant nous avons vu les outils nécessaires pour définir un élément, il convient maintenant de passer à l'utilisation de ces éléments pour discrétiser les formes intégrales, pour construire les matrices élémentaires.

Les différents opérateurs qui agissent sur u :

$$u = \langle N \rangle \{U_n\} = NU_n \quad (\text{I.34})$$

Chapitre I : Principe de la méthode des éléments finis

La perturbation de la fonction qui agisse sur u :

$$\delta u = \delta N U_n = N \delta U_n \quad (\text{I.35})$$

Par commodité nous écrivons :

$$\delta u = (N \delta U_n)^T = \delta U_n^T N^T = \langle \delta U_n \rangle \{N\} \quad (\text{I.36})$$

Finalement le calcul des matrices élémentaires revient à évaluer les intégrales des opérateurs précédant, pour cela l'intégration numérique est recommandée, elle consiste à remplacer l'intégrale exacte d'une fonction par une sommation de valeurs pondérées de cette fonction en des points spécifiques.

$$\int_{\Omega} f(x) dx = \sum_{i=1}^n w_i f(x_i) \quad (\text{I.37})$$

n : nombre de points de Gauss-Legendre.

w_i : valeurs de pondération correspondantes.

I.8 Assemblage et résolution

Une fois les matrices de rigidité et vecteurs forces élémentaires ($[K_e]$, $[F_e]$) sont calculées, il reste néanmoins à positionner chacun d'entre eux dans la matrice de rigidité $[K]$ et le vecteur $\{F\}$ caractérisant l'ensemble de la structure (global) et les sommer. Ce positionnement est mis en œuvre via une matrice de localisation $[L_i]$.

Autrement dit, on doit situer chaque composante de la matrice élémentaire dans la matrice globale, et les sommer avec les valeurs existant, qui est le résultat des contributions des autres matrices élémentaires.

Il ne reste qu'appliquer les conditions aux limites pour réduire le système global puis vient l'étape de résolution.

Chapitre I : Principe de la méthode des éléments finis

I.9 Convergence de la méthode des éléments finis

Divers types d'erreurs peuvent être introduites dans les différentes étapes de calculs par éléments finis ainsi que lors de la discrétisation du domaine, la précision dépend essentiellement du nombre d'éléments utilisés et du choix des fonctions d'approximation, donc la convergence peut être atteinte par l'augmentation du degré des polynômes d'interpolation (nombre de nœuds par éléments) ou du nombre d'éléments total (nous parlons du raffinement du maillage).

Conclusion

La MEF est une méthode numérique très puissante permettant, de déterminer de manière approximative, l'état d'équilibre d'un milieu continu.

Cette technique a comme avantage majeur la possibilité de développer un programme permettant de résoudre plusieurs types de problèmes avec peu de modifications. En particulier, toute forme complexe d'un domaine géométrique où un problème est bien posé avec toutes les conditions aux limites, peut être facilement traitée. Mais l'utilisation de cette méthode nécessite beaucoup d'outils mathématiques et informatiques.

Chapitre II : Rappels sur la mécanique des solides

Chapitre II : Rappel sur la mécanique des solides

II Introduction

Le deuxième chapitre concerne un rappel sur les bases de la mécanique des solides, tels que la notion de contraintes et de déformations, la relation entre contraintes et déformations et l'inverse, nous finirons par une petite présentation de l'énergie de déformation. Dans notre étude on se limite à l'élasticité linéaire, qui est une propriété physique d'un corps de lui rendre sa forme initiale après suppression des sollicitations. On tiendra compte de certaines hypothèses simplificatrices telles que :

- Le solide est supposé élastique, homogène et isotrope.
- Les déformations sont supposées très petites.
- La loi de comportement contrainte-déformation est supposée linéaire.

II.1 Les contraintes

II.1.1 Notion de contraintes [7]

Prenons un solide sollicité par des efforts (charges) externes voir (figure II.1), son état d'équilibre nous permet d'écrire :

$$\vec{F}_{1 \rightarrow 2} + \vec{F}_{2 \rightarrow 1} = \vec{0} \quad (\text{II.1})$$

Notre objectif est de déterminer les efforts internes du solide, une coupe de ce solide suivant un plan (P) normal à l'axe Ox , nous donne deux parties ([1]) et ([2]) limitées par une section S (figure II.2), nous définissons le vecteur contrainte au point M de dS comme étant la $\lim_{dS \rightarrow 0} \frac{d\vec{F}}{dS}$, $d\vec{F}$ est l'action exercée en M par la partie ([2]) sur la partie ([1]). Dans notre cas nous le nommons $\vec{\sigma}_1$.

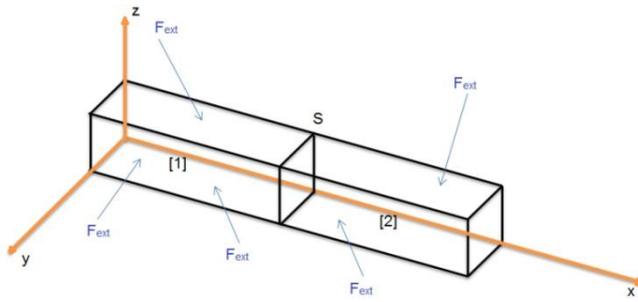


Figure II.1 : Équilibre d'un solide.

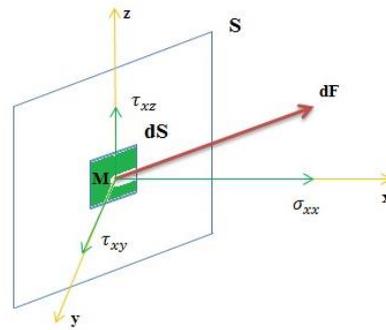


Figure II.2 : Vecteur contrainte.

Il peut être décomposé en 3 composantes (charge inclinée) illustré sur la (figure II.2), une normale à cette face et deux autres dans son plan (tangentes).

En répétant l'opération pour les plans $\{O; x, z\}$ ($\vec{\sigma}_2$) et $\{O; x, y\}$ ($\vec{\sigma}_3$), nous définissons les vecteurs contraintes :

$$\vec{\sigma}_1 = \begin{pmatrix} \sigma_{xx} \\ \tau_{xy} \\ \tau_{xz} \end{pmatrix} ; \vec{\sigma}_2 = \begin{pmatrix} \tau_{yx} \\ \sigma_{yy} \\ \tau_{yz} \end{pmatrix} ; \vec{\sigma}_3 = \begin{pmatrix} \tau_{zx} \\ \tau_{zy} \\ \sigma_{zz} \end{pmatrix} \quad (\text{II.2})$$

appelé aussi tenseur de contrainte $[\sigma]$. La symétrie de ce dernier $\sigma_{ij} = \sigma_{ji}$ nous permet d'écrire :

$$\tau_{xy} = \tau_{yx} ; \tau_{yz} = \tau_{zy} \text{ et } \tau_{xz} = \tau_{zx}.$$

Finalement les six valeurs caractérisant l'état de contraintes peuvent être regroupées dans un vecteur :

$$\{\sigma\} = \langle \sigma_{xx} \ \sigma_{yy} \ \sigma_{zz} \ \tau_{xy} \ \tau_{yz} \ \tau_{xz} \rangle^T \quad (\text{II.3})$$

II.1.2 Bilan d'équations internes

Considérant un volume élémentaire dV , soumis à des forces internes de volumes f_x^V, f_y^V, f_z^V voir (figure II.3), le bilan des forces sur ce volume dV suivant l'axe Ox nous donne :

$$\sum F_{/x} = 0$$

$$\frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} + f_x^V = 0$$

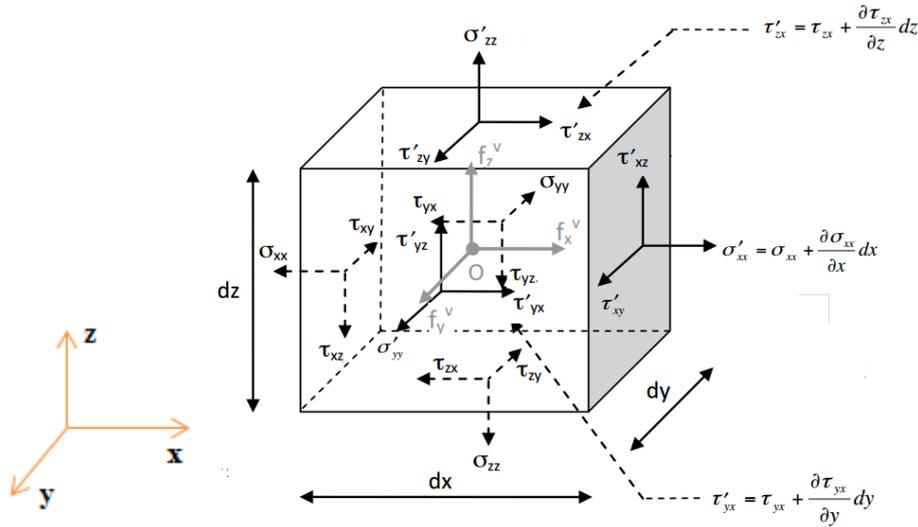


Figure II.3 : Equilibre de volume.

après l'extension de cette opération sur les deux autres axes :

$$\frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} + f_x^V = 0$$

$$\frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \sigma_{yy}}{\partial y} + \frac{\partial \tau_{zy}}{\partial z} + f_y^V = 0 \quad \Leftrightarrow \quad \frac{\partial \sigma_{ij}}{\partial x_j} + f_j = 0 \quad (\text{II.4})$$

$$\frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \sigma_{zz}}{\partial z} + f_z^V = 0$$

II.1.3 Bilan d'équations externes

Soit un tétraèdre infiniment petit $OACB$, sachant qu'une charge F^S de composante $\langle f_x^S, f_y^S, f_z^S \rangle^T$ appliquée sur la surface ABC notée ds (figure II.4).

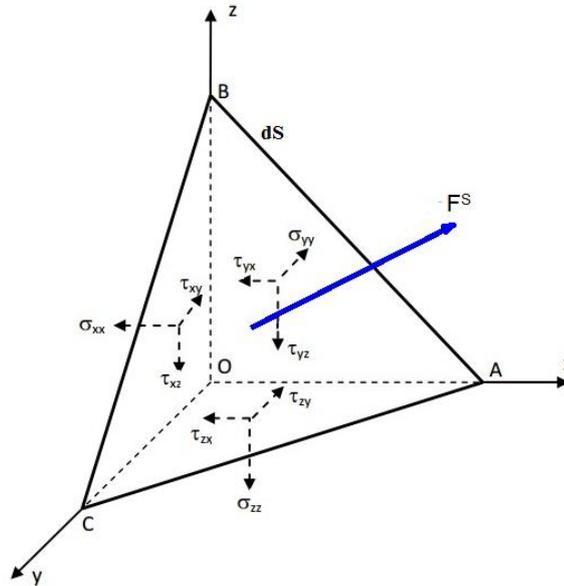


Figure II.4 : Equilibre de surface.

Les cosinus directeurs de la normale au plan ABC sont notés l , m et n , l'équilibre du tétraèdre se résume à :

$$f_x^S = l\sigma_{xx} + m\tau_{yx} + n\tau_{zx}$$

$$f_y^S = l\tau_{xy} + m\sigma_{yy} + n\tau_{zy} \quad (\text{II.5})$$

$$f_z^S = l\tau_{xz} + m\tau_{yz} + n\sigma_{zz}$$

pour l'expression matricielle :

$$\begin{Bmatrix} f_x^S \\ f_y^S \\ f_z^S \end{Bmatrix} = \begin{bmatrix} \sigma_{xx} & \tau_{yx} & \tau_{zx} \\ \tau_{xy} & \sigma_{yy} & \tau_{zy} \\ \tau_{xz} & \tau_{yz} & \sigma_{zz} \end{bmatrix} \begin{Bmatrix} l \\ m \\ n \end{Bmatrix} = [\sigma] \begin{Bmatrix} l \\ m \\ n \end{Bmatrix} \quad (\text{II.6})$$

Ces équations d'équilibres de surface correspondent en fait aux conditions aux limites du système d'équations aux dérivées partielles.

II.2 Les déformations

II.2.1 Déplacements [7]

Le mouvement d'un point P vers P' caractérisé par le vecteur \vec{p} peut-être décomposé dans un référentiel $\{O; x, y, z\}$ en 3 composantes u, v, w :

Chapitre II : Rappel sur la mécanique des solides

$$\vec{p} = u.\vec{i} + v.\vec{j} + w.\vec{k} \quad (\text{II.7})$$

Nous définissons par ailleurs la norme de déplacement comme étant :

$$\|\vec{p}\| = \sqrt{u^2 + v^2 + w^2} \quad (\text{II.8})$$

II.2.2 Relations entre déplacements et déformations

Les déformations peuvent être obtenues en considérant dans un premier temps un état plan de déformation dans le plan $\{O; x, y\}$ (figure II.5), en posant que :

$$\varepsilon_{xx} = \lim_{dx \rightarrow 0} \frac{a'c' - ac}{ac} = \lim_{dx \rightarrow 0} \frac{dx + u + \frac{\partial u}{\partial x} dx - (u + dx)}{dx} = \frac{\partial u}{\partial x} \quad (\text{II.9})$$

$$\varepsilon_{yy} = \lim_{dy \rightarrow 0} \frac{a'b' - ab}{ab} = \lim_{dy \rightarrow 0} \frac{dy + v + \frac{\partial v}{\partial y} dy - (v + dy)}{dy} = \frac{\partial v}{\partial y}$$

La distorsion angulaire est obtenue par :

$$\gamma_{xy} = \lim_{\substack{dx \rightarrow 0 \\ dy \rightarrow 0}} \left(\frac{\pi}{2} - \beta \right) = \frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \quad (\text{II.10})$$

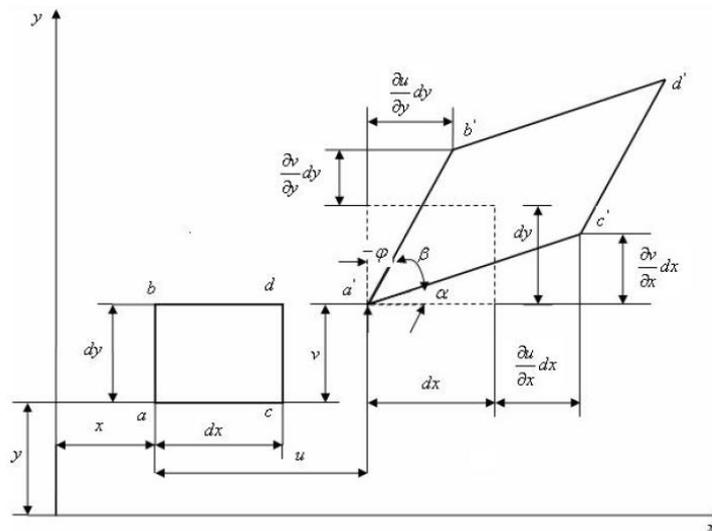


Figure II.5 : Déformations planes [7].

Si nous voulons généraliser cette approche bidimensionnelle aux deux autres plans nous obtiendrons :

Chapitre II : Rappel sur la mécanique des solides

$$\begin{aligned}\varepsilon_{xx} &= \frac{\partial u}{\partial x} & ; & \quad \gamma_{xy} = \frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \\ \varepsilon_{yy} &= \frac{\partial v}{\partial y} & ; & \quad \gamma_{yz} = \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \\ \varepsilon_{zz} &= \frac{\partial w}{\partial z} & ; & \quad \gamma_{zx} = \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\end{aligned}\tag{II.11}$$

la relation de la déformation en fonction des déplacements : $\varepsilon_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right)$ (II.12)

d'où l'expression générale des composantes du vecteur de déformation :

$$\{\varepsilon\} = \langle \varepsilon_{xx} \quad \varepsilon_{yy} \quad \varepsilon_{zz} \quad \gamma_{xy} \quad \gamma_{yz} \quad \gamma_{zx} \rangle^T \tag{II.13}$$

Nous pourrions également l'exprimer sous forme matricielle de cette façon :

$$\{\varepsilon\} = \begin{Bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \varepsilon_{zz} \\ \gamma_{xy} \\ \gamma_{yz} \\ \gamma_{zx} \end{Bmatrix} = \begin{bmatrix} \frac{\partial}{\partial x} & 0 & 0 \\ 0 & \frac{\partial}{\partial y} & 0 \\ 0 & 0 & \frac{\partial}{\partial z} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial z} & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} & 0 & \frac{\partial}{\partial x} \end{bmatrix} \begin{Bmatrix} u \\ v \\ w \end{Bmatrix} \tag{II.14}$$

II.3 Loi de comportement

Il existe une multitude de lois de comportement reliant les contraintes aux déformations, pour notre étude avec les hypothèses considérées dans l'introduction et pour un acier doux la loi de Hooke nous permet d'écrire [8] :

$$\sigma_{ij} = E_{ijkl} \cdot \varepsilon_{kl} \Rightarrow \sigma_{xx} = E \cdot \varepsilon_{xx} \tag{II.15}$$

liant linéairement contraintes et déformations. En plus de la déformation axiale, une déformation transversale est observée :

$$\text{Suivant } y : \quad \varepsilon_{yy} = -\nu \cdot \varepsilon_{xx} = -\nu \cdot \frac{\sigma_{xx}}{E} \tag{II.16}$$

Chapitre II : Rappel sur la mécanique des solides

$$\text{Suivant } z : \quad \varepsilon_{zz} = -\nu \cdot \varepsilon_{xx} = -\nu \cdot \frac{\sigma_{xx}}{E} \quad (\text{II.17})$$

ν correspond au coefficient de Poisson, il est strictement inférieur à 0.5.

II.3.1 Relations entre déformations et contraintes

Avec la même démarche et hypothèses précédentes, la relation tridimensionnelle entre déformations et contraintes normales peut être obtenue par la superposition des trois états d'équilibre x , y et z :

$$\begin{aligned} \varepsilon_{xx} &= \frac{1}{E} (\sigma_{xx} - \nu(\sigma_{yy} + \sigma_{zz})) \\ \varepsilon_{yy} &= \frac{1}{E} (\sigma_{yy} - \nu(\sigma_{xx} + \sigma_{zz})) \\ \varepsilon_{zz} &= \frac{1}{E} (\sigma_{zz} - \nu(\sigma_{xx} + \sigma_{yy})) \end{aligned} \quad (\text{II.18})$$

De plus, un essai de cisaillement pur nous montre qu'il existe des relations entre déformations et contraintes de cisaillement :

$$\tau_{xy} = G \cdot \gamma_{xy} ; \tau_{yz} = G \cdot \gamma_{yz} ; \tau_{xz} = G \cdot \gamma_{xz} \quad \text{avec } G = \frac{E}{2(1+\nu)} \quad (\text{II.19})$$

d'où la relation finale entre déformations et contraintes :

$$\{\varepsilon\} = [D] \cdot \{\sigma\} \quad \text{avec } [D] = \begin{bmatrix} \frac{1}{E} & \frac{-\nu}{E} & \frac{-\nu}{E} & 0 & 0 & 0 \\ \frac{-\nu}{E} & \frac{1}{E} & \frac{-\nu}{E} & 0 & 0 & 0 \\ \frac{-\nu}{E} & \frac{-\nu}{E} & \frac{1}{E} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{G} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{G} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{G} \end{bmatrix} \quad (\text{II.20})$$

II.3.2 Relations entre contraintes et déformations

En faisant l'inverse on déduit les contraintes à partir des déformations :

$$\begin{aligned} \sigma_{xx} &= \frac{E}{(1+\nu)(1-2\nu)} [\varepsilon_{xx} \cdot (1 - \nu) + \nu(\varepsilon_{yy} + \varepsilon_{zz})] \\ \sigma_{yy} &= \frac{E}{(1+\nu)(1-2\nu)} [\varepsilon_{yy} \cdot (1 - \nu) + \nu(\varepsilon_{xx} + \varepsilon_{zz})] \end{aligned}$$

$$\sigma_{zz} = \frac{E}{(1+\nu)(1-2\nu)} [\varepsilon_{zz} \cdot (1 - \nu) + \nu(\varepsilon_{xx} + \varepsilon_{zz})] \quad (\text{II.21})$$

$$\tau_{xy} = G \cdot \gamma_{xy}$$

$$\tau_{yz} = G \cdot \gamma_{yz}$$

$$\tau_{zx} = G \cdot \gamma_{zx}$$

soit sous forme matricielle :

$$\{\sigma\} = [H] \cdot \{\varepsilon\} \quad (\text{II.22})$$

avec :

$$[H] = \frac{E}{(1 + \nu)(1 - 2\nu)} \begin{bmatrix} 1 - \nu & \nu & \nu & 0 & 0 & 0 \\ \nu & 1 - \nu & \nu & 0 & 0 & 0 \\ \nu & \nu & 1 - \nu & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1 - 2\nu}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1 - 2\nu}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1 - 2\nu}{2} \end{bmatrix} = [D]^{-1}$$

$[H]$ est la matrice des propriétés matériaux, elle n'est pas définie pour $\nu = 0.5$

II.3.3 Etat plan de contraintes et de déformations

a) Hypothèse de contraintes planes [8]

L'hypothèse de la contrainte plane suppose que toutes les contraintes associées à l'axe transversal z sont nulles, nous avons donc :

$$\sigma_{zz} = \tau_{xz} = \tau_{yz} = 0 \text{ et } \gamma_{xz} = \gamma_{yz} = 0 \quad (\text{II.23})$$

La relation (II.18) permet d'établir :

$$\varepsilon_{zz} = -\frac{\nu}{E} (\sigma_{xx} + \sigma_{yy}) \quad (\text{II.24})$$

Nous pouvons écrire également :

Chapitre II : Rappel sur la mécanique des solides

$$\begin{Bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \tau_{xy} \end{Bmatrix} = \frac{E}{(1-\nu^2)} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \begin{Bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \gamma_{xy} \end{Bmatrix} \quad (\text{II.25})$$

Cette approche est appliquée aux éléments de faibles épaisseurs, les éléments membranes, plaques et coques suivront cette hypothèse de contrainte plane.

b) Hypothèse de déformations planes [8]

Hypothèse de déformation plane consiste à considérer que toutes les déformations associées à l'axe transversal z du solide, seront prises égal à zéro, donc :

$$\varepsilon_{zz} = \gamma_{xz} = \gamma_{yz} = 0 \quad \text{et} \quad \tau_{xz} = \tau_{yz} = 0. \quad (\text{II.26})$$

Remplaçons (II.26) dans (II.22) la relation contrainte-déformation devient pour la déformation plane :

$$\begin{Bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \tau_{xy} \end{Bmatrix} = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & 0 \\ \nu & 1-\nu & 0 \\ 0 & 0 & \frac{1-2\nu}{2} \end{bmatrix} \begin{Bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \gamma_{xy} \end{Bmatrix} \quad (\text{II.27})$$

De la relation (II.18), on déduit :
$$\sigma_{zz} = \nu(\sigma_{xx} + \sigma_{yy}) \quad (\text{II.28})$$

II.4 Energie de déformation élastique [7]

II.4.1 Cas du chargement uniaxial

Soit un volume élémentaire dV (figure II.6) ; l'énergie emmagasinée par ce volume soumis à un chargement uniaxial dans la direction longitudinal x est égale à :

$$dW = \frac{1}{2} \cdot F \cdot du \quad (\text{II.29})$$

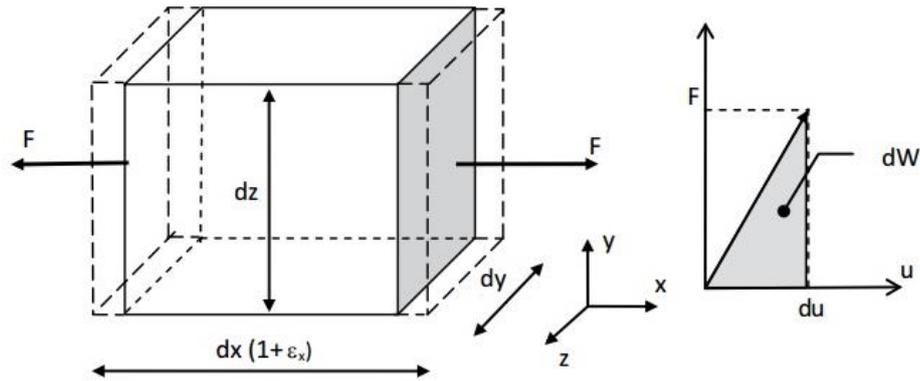


Figure II.6 : Energie de déformation uniaxial[7].

avec : $du = \varepsilon_{xx} \cdot dx$ et $F = \sigma_{xx} \cdot dydz$, dW peut être également exprimé sous forme :

$$dW = \frac{1}{2} \cdot \sigma_{xx} \cdot \varepsilon_{xx} \cdot dx dy dz \quad (\text{II.30})$$

Pour tout le volume d'un solide, l'expression de l'énergie de déformation devient :

$$W = \frac{1}{2} \int_V \sigma_{xx} \cdot \varepsilon_{xx} \cdot dV \quad (\text{II.31})$$

II.4.2 Cas général

Pour les deux autres axes y et z toujours dans le cas uniaxial nous obtiendrons :

$$W = \frac{1}{2} \int_V (\sigma_{xx} \cdot \varepsilon_{xx} + \sigma_{yy} \cdot \varepsilon_{yy} + \sigma_{zz} \cdot \varepsilon_{zz}) \cdot dV \quad (\text{II.32})$$

La démarche est la même pour les cisaillements :

$$W = \frac{1}{2} \int_V \tau_{xy} \cdot \gamma_{xy} \cdot dV \quad (\text{II.33})$$

L'énergie de déformation élastique deviendra alors :

$$W = \frac{1}{2} \int_V (\sigma_{xx} \cdot \varepsilon_{xx} + \sigma_{yy} \cdot \varepsilon_{yy} + \sigma_{zz} \cdot \varepsilon_{zz} + \tau_{xy} \cdot \gamma_{xy} + \tau_{yz} \cdot \gamma_{yz} + \tau_{zx} \cdot \gamma_{zx}) \cdot dV \quad (\text{II.34})$$

Sous forme matricielle :

$$W = \frac{1}{2} \int_V \{\varepsilon\}^T \cdot \{\sigma\} \cdot dV \quad (\text{II.35})$$

Chapitre III :
Formulation des
éléments barre, poutre
et membrane par la
MEF

III Introduction

Le troisième chapitre est l'objectif principal de notre projet, qui consiste à formuler les éléments de structures (barre, poutre et membrane) par la méthode des éléments finis puis élaborer un programme pour leurs calcul, nous commençons par l'élément barre qui est soumis uniquement à des charges axiales, puis vient l'élément poutre qui travaille généralement en flexion, finalement l'élément membrane qui est chargé uniquement dans son plan.

III.1 Élément barre à deux nœuds

Une barre est un élément à 2 nœuds comportant un seul degré de liberté dans son repère local et deux degrés de liberté dans le repère global. Ses caractéristiques géométriques et matérielles se résument à une section constante A , une longueur L et un module d'élasticité longitudinal E . Fonctionnant en traction et en compression uniquement (l'effort normal est positif dans le cas de la traction).

III.1.1 Les étapes d'analyse par éléments finis d'une barre [4,7,9]

III.1.1.1 Equations gouvernantes

Pour formuler cet élément, nous considérons une barre soumise à une traction $F(x)$ variante de F_0 et F comme le montre la (figure III.1).

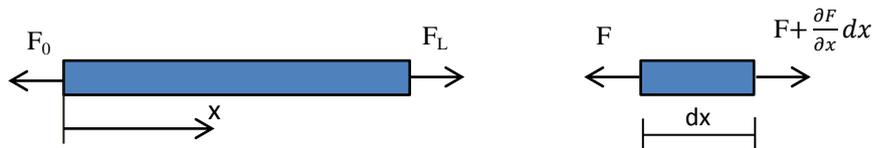


Figure III.1 : Equilibre élémentaire d'une barre.

Un tronçon infinitésimal de longueur dx situé à la coordonnée x le long de la barre est en équilibre statique :

$$\sum F = 0$$

$$\left[F + \frac{\partial F}{\partial x} dx \right] - F = 0 \quad (\text{III.1})$$

$$\frac{F}{A} = \sigma_x = E \varepsilon_x \quad (\text{III.2})$$

la déformation étant la dérivée du déplacement :

$$\varepsilon_x = \frac{\partial u}{\partial x} \quad (\text{III.3})$$

où u désigne le déplacement longitudinal, x est la coordonnée. La loi de Hooke nous permet d'écrire :

La substitution de (III.1) dans (III.2) nous donne l'expression de la charge F en fonction du déplacement u :

$$F = EA \frac{\partial u}{\partial x} \quad (\text{III.4})$$

D'où l'équation différentielle de l'équilibre de l'élément dx qui s'obtient en remplaçant (III.4) dans (III.1) :

$$\frac{\partial}{\partial x} (EA \frac{\partial u}{\partial x}) dx = 0 \quad (\text{III.5})$$

Les conditions aux limites de cette équation dépendent de l'encastrement ou des déplacements imposés pour u et du chargement aux nœuds pour les dérivées de u .

III.1.1.2 Formulation de l'élément

En prenant δu comme fonction de pondération, la formulation intégrale forte de l'équation (III.5) s'écrit :

$$\int_0^L \delta u \left[\frac{\partial}{\partial x} (EA \frac{\partial u}{\partial x}) \right] dx = 0 \quad (\text{III.6})$$

La formulation faible s'écrit en prenant l'intégration par parties de l'équation (III.6):

$$\int_0^L \frac{\delta \partial u}{\partial x} EA \frac{\partial u}{\partial x} dx = [\delta u AE \frac{\partial u}{\partial x}]_0^L \quad (\text{III.7})$$

Le deuxième terme représente la différence des forces appliquées aux extrémités (nœuds) de la barre : $\delta u|_{x=L} \cdot F_L - \delta u|_{x=0} \cdot F_0$. Pour la discrétisation de l'équation (III.7) nous prendrons des fonctions de forme linéaires avec $x_1 = 0$ et $x_2 = L$ leurs expressions sont :

$$N(x) = \left\langle \frac{x-L}{0-L} \quad \frac{x-0}{L-0} \right\rangle = \frac{1}{L} \langle L-x \quad x \rangle ; dN(x) = \frac{1}{L} \langle -1 \quad 1 \rangle \quad (\text{III.8})$$

Ainsi après remplacement de différentes expressions de la forme faible nous obtiendrons :

$$\int_0^L \delta U^T dN^T EA. dNU dx + [\delta U^T N^T AE. NU]_0^L = 0 \quad (\text{III.9})$$

$$\delta U^T dN^T = \langle \delta u_0 \quad \delta u_L \rangle \begin{Bmatrix} -1/L \\ 1/L \end{Bmatrix} \quad dNU = \frac{1}{L} \langle -1 \quad 1 \rangle \begin{Bmatrix} u_0 \\ u_L \end{Bmatrix}$$

$$\delta u|_{x=0} = \langle \delta u_0 \quad \delta u_L \rangle \begin{Bmatrix} 1 \\ 0 \end{Bmatrix} \quad \delta u|_{x=L} = \langle \delta u_0 \quad \delta u_L \rangle \begin{Bmatrix} 0 \\ 1 \end{Bmatrix}$$

Après simplification de $\langle \delta u_0 \quad \delta u_L \rangle$ nous aboutirons :

$$\int_0^L \begin{Bmatrix} -1/L \\ 1/L \end{Bmatrix} EA \langle -\frac{1}{L} \quad \frac{1}{L} \rangle dx \begin{Bmatrix} u_0 \\ u_L \end{Bmatrix} = \begin{Bmatrix} -F_0 \\ F_L \end{Bmatrix} \quad (\text{III.10})$$

Soit sous forme matricielle : $K_e U_e = F_e$

$$\begin{aligned} K_e &= \int_0^L \begin{Bmatrix} -1/L \\ 1/L \end{Bmatrix} EA \langle -\frac{1}{L} \quad \frac{1}{L} \rangle dx \\ &= \frac{1}{L^2} \int_0^L EA \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} dx = \frac{EA}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \end{aligned} \quad (\text{III.11})$$

$$F_e = \begin{Bmatrix} 0 \\ 1 \end{Bmatrix} F_L - \begin{Bmatrix} 1 \\ 0 \end{Bmatrix} F_0 = \begin{Bmatrix} -F_0 \\ F_L \end{Bmatrix} \quad (\text{III.12})$$

K_e représente la matrice de rigidité élémentaire et F_e est le vecteur de chargement de l'élément.

III.1.2 Treillis plans [4 ,9]

Les structures à treillis sont constituées par des assemblages de barres liées entre elles de telle sorte que le chargement extérieur soit repris uniquement par des forces axiales dans les barres. La (figure III.2) montre un exemple de système à treillis composée d'un assemblage de 17 barres et soumis à un chargement de trois forces.

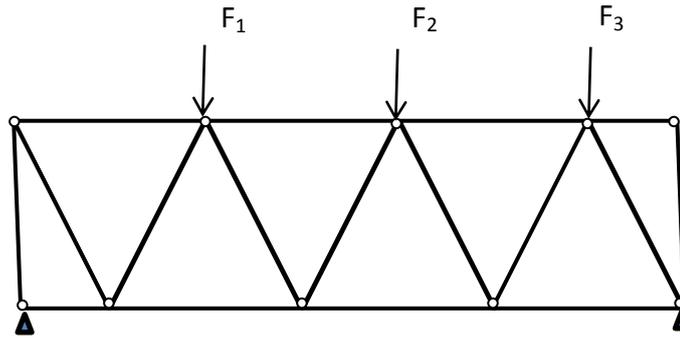


Figure III.2 : Système à treillis.

Les barres composant ce système subissent deux déplacements à leurs extrémités, une composante horizontale et une autre verticale. Cependant seul le déplacement axial à la barre donne naissance à la force. Ainsi les matrices élémentaires d'une barre bidimensionnelle (figure III.3) deviennent des matrices 4x4 puisque le vecteur des déplacements élémentaires devient :

$$U_e = \langle u_1 \ v_1 \ u_2 \ v_2 \rangle^T \quad (\text{III.13})$$

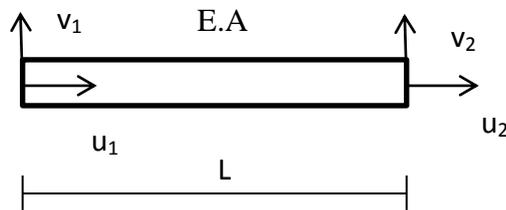


Figure III.3 : Élément barre bidimensionnelle.

Toutes les composantes associées au déplacement v de la matrice de rigidité sont nulles lors d'un effort horizontal, la matrice s'écrit ainsi :

$$K'_e = \frac{EA}{L} \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (\text{III.14})$$

L'expression d'une matrice élémentaire telle qu'elle est écrite par l'équation (III.14) est valable dans un système d'axes confondu avec l'axe longitudinal de l'élément de barre, en pratique la barre peut être inclinée par rapport aux axes de référence, il est nécessaire de faire une rotation d'axes pour venir aux axes de la barre. Comme le montre la (figure III.4)

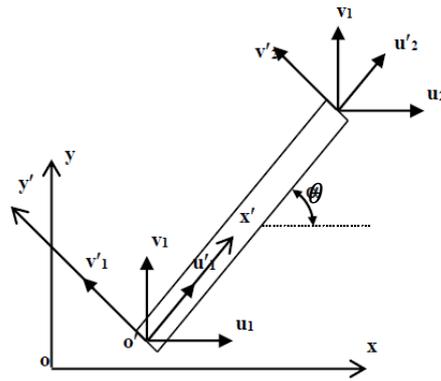


Figure III.4 : Elément barre incliné [6].

nous pouvons faire un passage d'un système local à un système global à l'aide de la matrice de rotation (passage) R :

$$\begin{Bmatrix} u'_1 \\ v'_1 \end{Bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{Bmatrix} u_1 \\ v_1 \end{Bmatrix}; \begin{Bmatrix} u_1 \\ v_1 \end{Bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{Bmatrix} u'_1 \\ v'_1 \end{Bmatrix} \quad (\text{III.15})$$

Pour la totalité des déplacements élémentaires :

$$\begin{Bmatrix} u'_1 \\ v'_1 \\ u'_2 \\ v'_2 \end{Bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & \cos\theta & \sin\theta \\ 0 & 0 & -\sin\theta & \cos\theta \end{bmatrix} \begin{Bmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \end{Bmatrix} \Rightarrow \{U'_e\} = [R]\{U_e\} \quad (\text{III.16})$$

Le repère lié à la barre nous permet d'écrire : $[K'_e]\{U'_e\} = \{F'_e\}$, la relation entre les deux vecteurs force est $\{F'_e\} = [R]\{F_e\}$ où bien $\{F_e\} = [R]^T\{F'_e\} = [R]^T[K'_e]\{U'_e\} = [R]^T[K'_e][R]\{U_e\}$.

finalement : $\{F_e\} = [K_e]\{U_e\}$, soit :

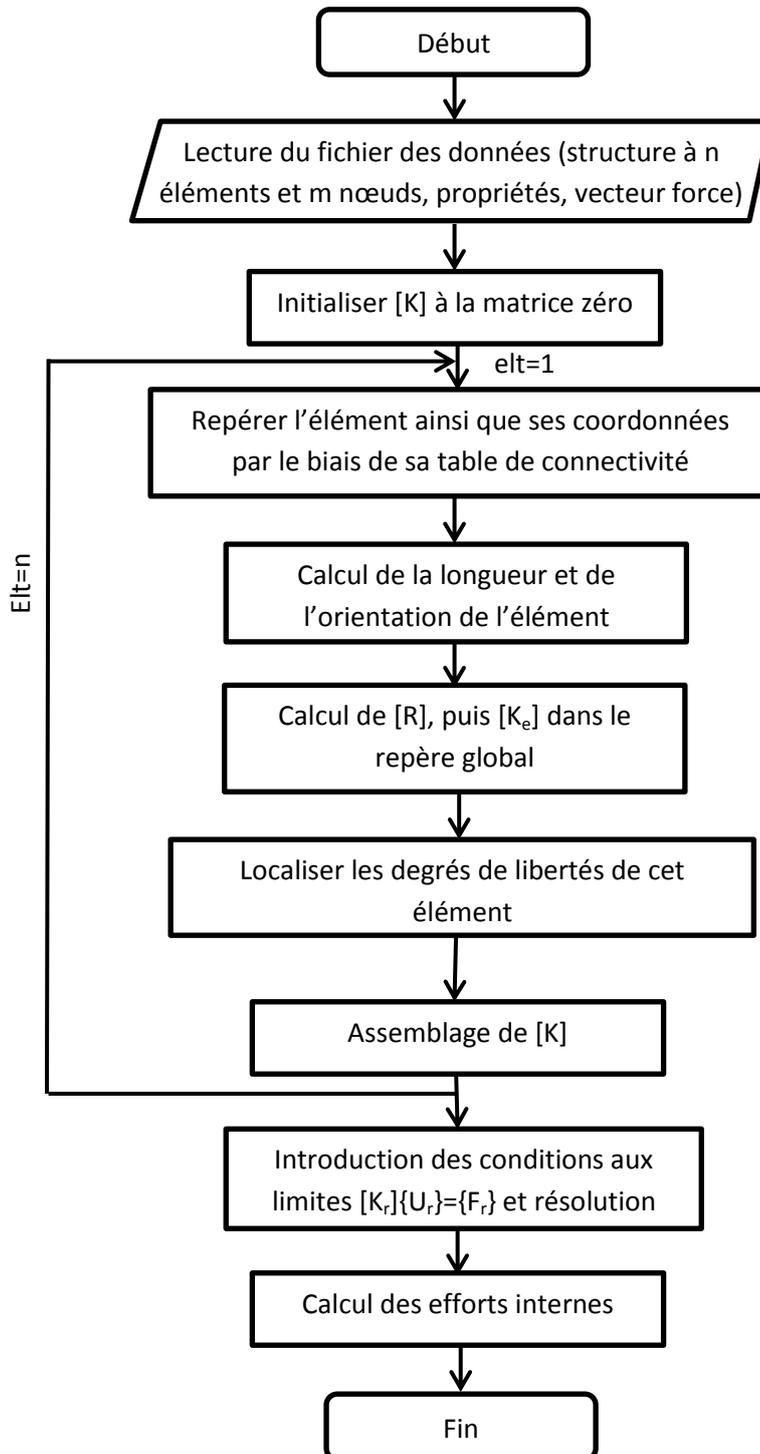
$$[K_e] = [R]^T[K'_e][R] \quad (\text{III.17})$$

$[K_e] =$

$$\begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & \cos\theta & -\sin\theta \\ 0 & 0 & \sin\theta & \cos\theta \end{bmatrix} \frac{AE}{L} \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & \cos\theta & \sin\theta \\ 0 & 0 & -\sin\theta & \cos\theta \end{bmatrix}$$

$$= \frac{AE}{L} \begin{bmatrix} \cos^2\theta & \cos\theta.\sin\theta & -\cos^2\theta & -\cos\theta.\sin\theta \\ \cos\theta.\sin\theta & \sin^2\theta & -\cos\theta.\sin\theta & -\sin^2\theta \\ -\cos^2\theta & -\cos\theta.\sin\theta & \cos^2\theta & \cos\theta.\sin\theta \\ -\cos\theta.\sin\theta & -\sin^2\theta & \cos\theta.\sin\theta & \sin^2\theta \end{bmatrix} \quad (\text{III.18})$$

III.1.3 Organigramme de résolution de l'élément barre



III.2 Élément poutre à deux nœuds

III.2.1 Définition

Une poutre est un solide engendré par un ensemble de surface S dites « sections droites », dont les deux dimensions sont petites par rapport à la troisième, on définit la ligne moyenne comme étant la succession des centres de gravités de chaque section droite, une poutre travail généralement en flexion.

III.2.2 Les étapes d'analyse d'une poutre à deux nœuds [4,7,9]

III.2.2.1 Equations gouvernantes

a) comportement axial

Dans ce cas une poutre ce comporte exactement comme une barre.

b) comportement flexionnel

Considérant une poutre de section A et de longueur L soumise à un chargement $q(x)$ qui varie tout au long de son axe longitudinal (figure III.5) :

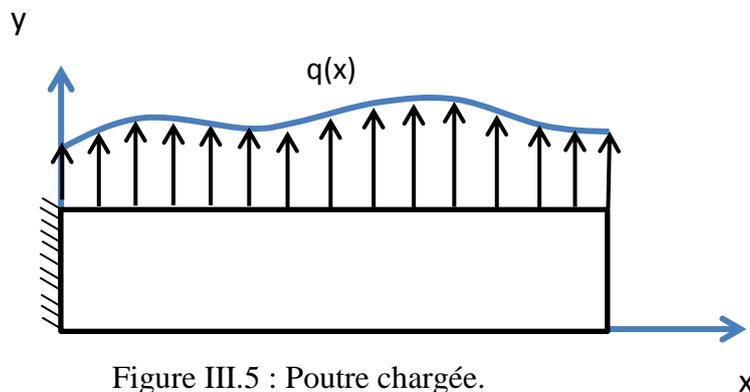


Figure III.5 : Poutre chargée.

Analysons un élément dx de la poutre délimité par deux sections voisines, la poutre fléchit sous l'effet du chargement et se déplace verticalement d'un déplacement $v(x)$, on se base sur la théorie de Bernoulli qui décrit que les sections droites restent planes et perpendiculaires à la ligne moyenne voir (figure III.6).

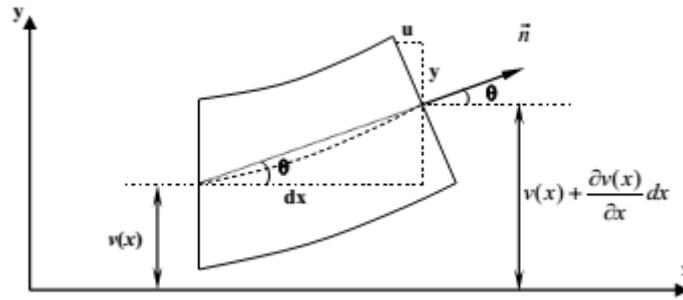


Figure III.6 Déformation d'une poutre.

La rotation de la section déformée est la tangente de la ligne moyenne courbée :

$$\theta = \operatorname{tg} \theta = \frac{v(x) + \frac{\partial v(x)}{\partial x} dx - v(x)}{dx} = \frac{\partial v}{\partial x} \quad (\text{III.19})$$

Le déplacement horizontal u causé par la rotation s'écrit :

$$u = -\theta y = -y \frac{\partial v}{\partial x} \quad (\text{III.20})$$

La loi de Hooke nous donne la répartition des contraintes le long de la section :

$$\sigma_x = E \varepsilon_x = E \frac{\partial u}{\partial x} = -y E \frac{\partial^2 v}{\partial x^2} \quad (\text{III.21})$$

Le moment créé par ces contraintes équilibre le moment de flexion M créé par le chargement extérieur :

$$M = - \int_S \sigma_x \cdot y \cdot dS = E \frac{\partial^2 v}{\partial x^2} \int_S y^2 dS = EI \frac{\partial^2 v}{\partial x^2} \quad (\text{III.22})$$

I est le moment d'inertie par rapport à l'axe z .

Considérant l'équilibre statique d'un élément dx de la poutre, la somme des moments par rapport à son centre de gravité donne :

$$-M + T \frac{dx}{2} + (T + dT) \frac{dx}{2} + (M + dM) = 0 \quad (\text{III.23})$$

$T dx + dT \frac{dx}{2} + dM = 0$, en négligeant le second terme, on obtient la relation entre l'effort tranchant et le moment fléchissant :

$$T = - \frac{dM}{dx} = - \frac{\partial}{\partial x} \left(EI \frac{\partial^2 v}{\partial x^2} \right) \quad (\text{III.24})$$

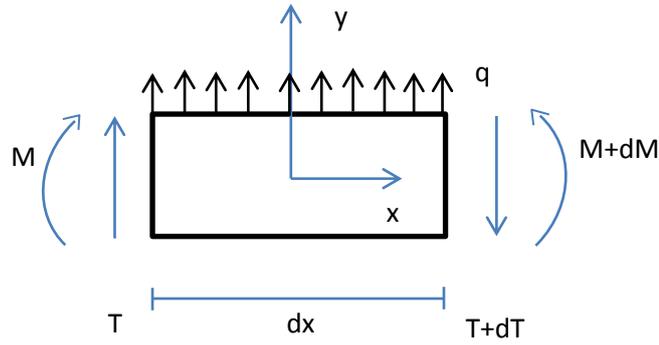


Figure III.7 : Equilibre statique.

Passant à l'équilibre des forces :

$$qdx + (T + dT) - T = 0 \quad (\text{III.25})$$

Qui donne la relation entre le chargement q et l'effort tranchant T et relie le chargement au déplacement v :

$$q(x) - \frac{dT}{dx} = \frac{\partial^2}{\partial x^2} (EI \frac{\partial^2 v}{\partial x^2}) \quad (\text{III.26})$$

L'équation (III.26) est dite équation d'Euler-Bernoulli pour la flexion des poutres.

III.2.2.2 Formulation intégrale

En prenant δv la fonction poids, la formulation intégrale forte de l'équation (III.26) s'écrit comme suite :

$$\int_0^L \delta v \left[\frac{\partial^2}{\partial x^2} (EI \frac{\partial^2 v}{\partial x^2}) \right] dx = \int_0^L \delta v q(x) dx \quad (\text{III.27})$$

La forme intégrale faible s'obtient avec deux intégrations par parties :

$$\int_0^L \delta v \left[\frac{\partial^2}{\partial x^2} (EI \frac{\partial^2 v}{\partial x^2}) \right] dx = \left[\delta v \frac{\partial}{\partial x} (EI \frac{\partial^2 v}{\partial x^2}) \right]_0^L - \int_0^L \frac{\partial \delta v}{\partial x} \frac{\partial}{\partial x} (EI \frac{\partial^2 v}{\partial x^2}) dx \quad (\text{III.28a})$$

$$\int_0^L \frac{\partial \delta v}{\partial x} \frac{\partial}{\partial x} (EI \frac{\partial^2 v}{\partial x^2}) dx = \left[\frac{\partial \delta v}{\partial x} EI \frac{\partial^2 v}{\partial x^2} \right]_0^L - \int_0^L \frac{\partial^2 \delta v}{\partial x^2} (EI \frac{\partial^2 v}{\partial x^2}) dx \quad (\text{III.28b})$$

Les premiers termes représentent la différence des chargements en force (T_0 et T_L) et en moment (M_0 et M_L) appliqués aux deux extrémités de la poutre, on peut remplacer aussi : $\frac{\delta \partial v}{\partial x} = \delta \theta$

On écrit les conditions aux limites comme suite :

$$\left[\frac{\partial \delta v}{\partial x} EI \frac{\partial^2 v}{\partial x^2} \right]_0^L = \delta \theta|_{x=0} M_0 - \delta \theta|_{x=L} M_L \quad (\text{III.29a})$$

$$\left[\delta v \frac{\partial}{\partial x} (EI \frac{\partial^2 v}{\partial x^2}) \right]_0^L = \delta v|_{x=L} T_L - \delta v|_{x=0} T_0 \quad (\text{III.29b})$$

En substituant (III.29) dans (III.28) et le résultat dans (III.27) on obtient l'expression de la forme intégral faible :

$$\int_0^L \frac{\partial^2 \delta v}{\partial x^2} (EI \frac{\partial^2 v}{\partial x^2}) dx + \delta \theta|_{x=L} M_L + \delta v|_{x=L} T_L - \delta \theta|_{x=0} M_0 - \delta v|_{x=0} T_0 = \int_0^L \delta v q(x) dx \quad (\text{III.30})$$

III.2.2.3 Discrétisation

Nous prendrons un élément à deux nœuds à deux degrés de libertés par nœud (figure III.8), dans le but d'assurer la continuité des déplacements ainsi que leurs dérivées qui sont les rotations. Le nombre de degrés de liberté atteint ainsi quatre et le polynôme d'interpolation doit être cubique. Le vecteur des déplacements et rotations élémentaires s'écrit donc comme suit :

$$U_n = \langle v_1 \ \theta_1 \ v_2 \ \theta_2 \rangle^T \quad (\text{III.31})$$


Figure III.8 : Élément poutre à deux nœuds.

Les déplacements et les rotations le long de la poutre sont exprimés sous forme polynomiale de degré 3 de la forme suivante:

$$v(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 = \langle 1 \ x \ x^2 \ x^3 \rangle \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{pmatrix} \quad (\text{III.32})$$

L'évaluation des polynômes aux nœuds donne :

$$v(0) = v_1$$

$$\theta(0) = v'(0) = \theta_1$$

$$v(L) = v_1 + a_1L + a_2L^2 + a_3L^3 \quad \theta(L) = v'(L) = \theta_1 + 2a_2L + 3a_3L^2$$

Le vecteur des déplacements nodaux s'écrit :

$$\begin{pmatrix} v_1 \\ \theta_1 \\ v_2 \\ \theta_2 \end{pmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & L & L^2 & L^3 \\ 0 & 2 & 2L & 3L^2 \end{bmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{pmatrix} = [R] \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{pmatrix} \quad (\text{III.33})$$

En injectant (III.33) dans (III.32)

$$v(x) = \langle 1 \quad x \quad x^2 \quad x^3 \rangle [R]^{-1} \begin{pmatrix} v_1 \\ \theta_1 \\ v_2 \\ \theta_2 \end{pmatrix} \quad (\text{III.34})$$

$$v(x) = \langle 1 \quad x \quad x^2 \quad x^3 \rangle \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -3/L^2 & -2/L & 3/L^2 & -1/L \\ 2/L^3 & 1/L^2 & -2/L^3 & 1/L^2 \end{bmatrix} \begin{pmatrix} v_1 \\ \theta_1 \\ v_2 \\ \theta_2 \end{pmatrix} \quad (\text{III.35})$$

Finalement :

$$v(x) = \langle 1 - \frac{3x^2}{L^2} + \frac{2x^3}{L^3} \quad x - \frac{2x^2}{L} + \frac{x^3}{L^2} \quad \frac{3x^3}{L^2} - \frac{2x^3}{L^3} \quad -\frac{x^2}{L} + \frac{x^3}{L^2} \rangle \begin{pmatrix} v_1 \\ \theta_1 \\ v_2 \\ \theta_2 \end{pmatrix} \quad (\text{III.36})$$

$$v(x) = N_1v_1 + N_2\theta_1 + N_3v_2 + N_4\theta_2 \quad (\text{III.37})$$

Les fonctions de forme N_i sont dites de type Hermite.

III.2.2.4 Matrices élémentaires

Il nous reste qu'à remplacer le déplacement par son approximation :

$$\delta v = \delta U_n^T N^T \quad \delta \theta = U_n^T dN^T \quad \text{avec} \quad \delta U_n^T = \langle \delta v_1 \quad \delta \theta_1 \quad \delta v_2 \quad \delta \theta_2 \rangle$$

$$\frac{\partial^2 v}{\partial x^2} = \frac{d^2 N}{dx^2} U_n \quad \text{avec} \quad \frac{d^2 N}{dx^2} = \frac{1}{L^3} \langle 12x - 6L \quad 6Lx - 4L^2 \quad 6L - 12x \quad 6Lx - 2L^2 \rangle$$

Le premier terme de l'équation (III.30) devient :

$$\int_0^L \frac{\partial^2 \delta v}{\partial x^2} (EI \frac{\partial^2 v}{\partial x^2}) dx = \delta U_n^T \int_0^L \frac{d^2 N^T}{dx^2} EI \frac{d^2 N}{dx^2} dx U_n \quad (\text{III.38})$$

Pour les conditions aux limites, ils correspondent au chargement extérieur appliqué aux nœuds, le vecteur élémentaire F_{nod} des forces et des moments concentrés aux nœuds s'écrit comme suit :

$$\{F_{nod}\} = \langle -T_0 \quad -M_0 \quad T_L \quad M_L \rangle^T \quad (III.39)$$

Le chargement F_e sur l'élément poutre correspond au second terme de la forme intégrale de l'équation (III.30) :

$$\int_0^L \delta v q(x) dx = \delta U_n^T \int_0^L N^T q(x) dx ; F_e = \delta U_n^T \int_0^L N^T q(x) dx \quad (III.40)$$

On remplace ce chargement réparti q par deux forces équivalentes concentrées aux deux extrémités. L'expression de F_{rep} est :

$$q(x) = q_0 + (q_L - q_0) \frac{x}{L}$$

$$F_{rep} = \delta U_n^T \int_0^L N^T (q_0 + (q_L - q_0) \frac{x}{L}) dx \quad (III.41)$$

Après intégration nous obtenons :

$$\{F_{rep}\} = \frac{L}{60} \langle (9q_L + 21q_0) \quad L(2q_L + 3q_0) \quad (21q_L + 9q_0) \quad -L(3q_L + 2q_0) \rangle^T \quad (III.42)$$

La simplification de δU_n^T pour tous les termes nous conduit à définir la matrice de rigidité élémentaire à partir de l'équation (III.36) :

$$K_e = \int_0^L \frac{d^2 N^T}{dx^2} EI \frac{d^2 N}{dx^2} \quad (III.43)$$

Son expression finale :

$$[K_e] = \frac{EI}{L^3} \begin{bmatrix} 12 & 6L & -12 & 6L \\ 6L & 4L^2 & -6L & 2L^2 \\ -12 & -6L & 12 & -6L \\ 6L & 2L^2 & -6L & 4L^2 \end{bmatrix} \quad (III.44)$$

Finalement en incluant les efforts axiaux, l'élément poutre à deux nœuds est caractérisé par :

- le vecteur élémentaire $\{F_{nod}\}$:

$$\{F_{nod}\} = \langle -N_0 \quad -T_0 \quad -M_0 \quad N_T \quad T_L \quad M_L \rangle^T \quad (III.45)$$

- le vecteur élémentaire F_{rep} :

$$\{F_{rep}\} = \frac{L}{60} \langle (0 \quad 9q_L + 21q_0) \quad L(2q_L + 3q_0) \quad 0 \quad (21q_L + 9q_0) \quad -L(3q_L + 2q_0) \rangle^T$$

(III.46)

- la matrice de rigidité élémentaire K_e :

$$[K_e] = \begin{bmatrix} \frac{EA}{L} & 0 & 0 & -\frac{EA}{L} & 0 & 0 \\ 0 & 12\frac{EI}{L^3} & 6\frac{EI}{L^2} & 0 & -12\frac{EI}{L^3} & 6\frac{EI}{L^2} \\ 0 & 6\frac{EI}{L^2} & 4\frac{EI}{L} & 0 & -6\frac{EI}{L^2} & \frac{2EI}{L} \\ -\frac{EA}{L} & 0 & 0 & \frac{EA}{L} & 0 & 0 \\ 0 & -12\frac{EI}{L^3} & -6\frac{EI}{L^2} & 0 & 12\frac{EI}{L^3} & -6\frac{EI}{L^2} \\ 0 & 6\frac{EI}{L^2} & \frac{2EI}{L} & 0 & -6\frac{EI}{L^2} & \frac{4EI}{L} \end{bmatrix} \quad (III.47)$$

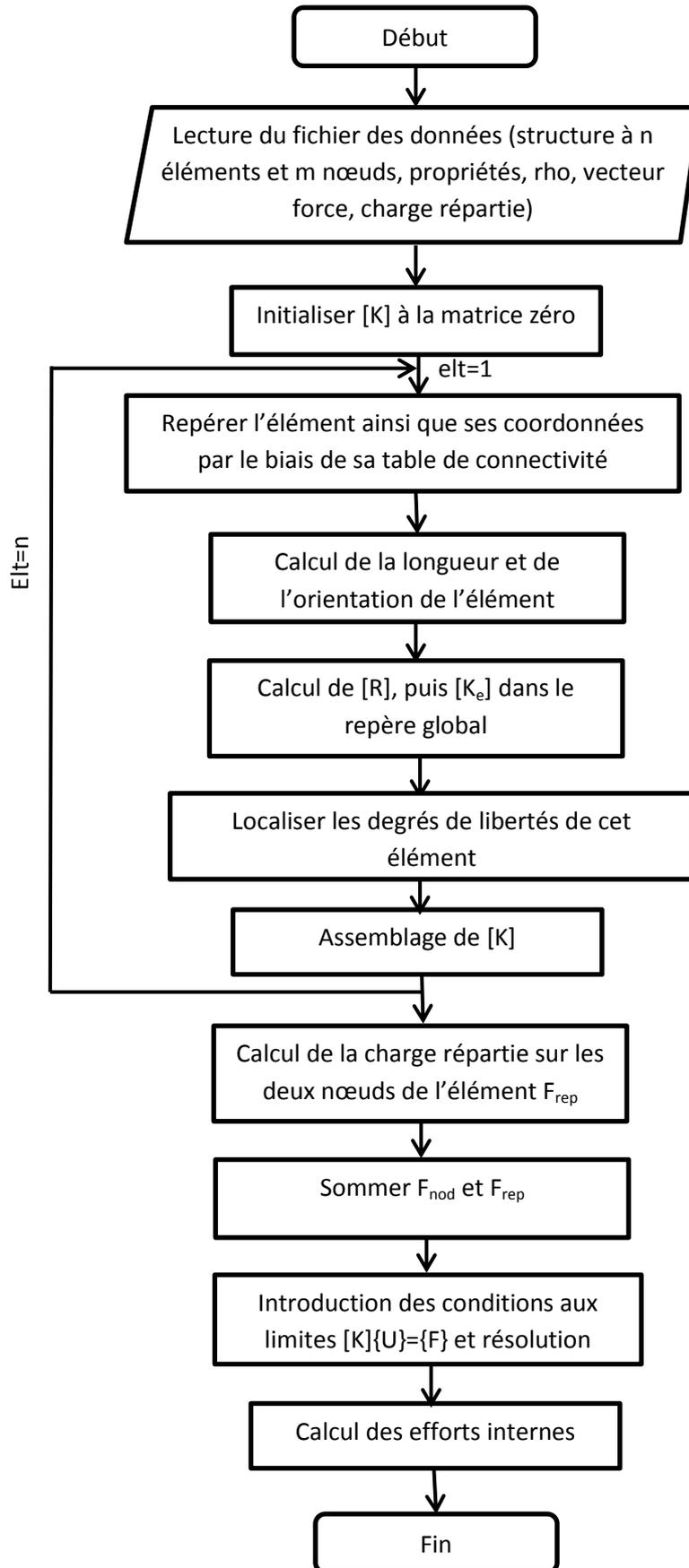
De la même manière avec les barres nous pouvons écrire la matrice de rigidité dans le repère global à l'aide de la matrice de rotation (changement de base).

$$[K_{e_g}] = [R]^T [K_e] [R] \quad (III.48)$$

avec :

$$[R] = \begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos\theta & \sin\theta & 0 \\ 0 & 0 & 0 & -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

III.2.3 Organigramme de résolution de l'élément poutre



III.3 Elément membrane

III.3.1 Définition

Les éléments membranes sont employés pour analyser des structures chargés uniquement dans leurs plans. Leurs nœuds possèdent deux degrés de liberté u et v , elles sont définies par une longueur a et une largeur b et une épaisseur constante notée t voir (figure III.9) [10].

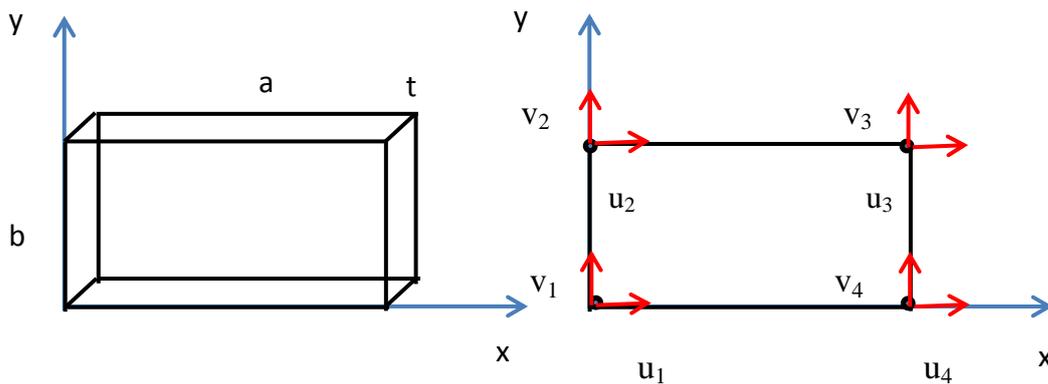


Figure III.9 : Elément membrane.

III.3.2 Etapes d'analyse de l'élément membrane [7,10]

III.3.2.1 Equation gouvernante

La relation contraintes-déformations (II.25) page 27 du deuxième chapitre nous permet d'établir l'état de contrainte plan d'un solide :

$$\begin{Bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \tau_{xy} \end{Bmatrix} = \frac{E}{(1-\nu^2)} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \begin{Bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \gamma_{xy} \end{Bmatrix}$$

En exprimant les déformations par les déplacements la relation devient :

$$\begin{Bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \tau_{xy} \end{Bmatrix} = \frac{E}{(1-\nu^2)} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \begin{bmatrix} \frac{\partial}{\partial x} & 0 & 0 \\ 0 & \frac{\partial}{\partial y} & 0 \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & 0 \end{bmatrix} \begin{Bmatrix} u \\ v \end{Bmatrix}$$

III.3.2.2 Fonctions de forme

les déplacements sont définis par des fonctions polynomiales :

$$u(x) = a_1 + a_2x + a_3y + a_4xy \quad (\text{III.49a})$$

$$v(x) = a_5 + a_6x + a_7y + a_8xy \quad (\text{III.49b})$$

soit sous forme vectorielle :

$$\begin{Bmatrix} u(x) \\ v(x) \end{Bmatrix} = \begin{bmatrix} 1 & x & y & xy & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & x & y & xy \end{bmatrix} \begin{Bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \\ a_8 \end{Bmatrix} \quad (\text{III.50})$$

La relation $u = \langle P_n \rangle [P_n]^{-1} U_n$ nous donne les fonctions de forme :

$$N_1(x, y) = \left(1 - \frac{x}{a}\right) \left(1 - \frac{y}{b}\right) ; \quad N_2(x, y) = \frac{x}{a} \left(1 - \frac{y}{b}\right) \quad (\text{III.51a})$$

$$N_3(x, y) = \frac{xy}{ab} ; \quad N_4(x, y) = \left(1 - \frac{x}{a}\right) \frac{y}{b} \quad (\text{III.51b})$$

III.3.2.3 Matrice de rigidité et vecteur charge

La relation (I.9) page 5 du premier chapitre permet d'établir la matrice de rigidité élémentaire en tenant compte de l'épaisseur constante t :

$$K = t \int_a^b \int_a^b [B]^T [H] [B] dx dy \quad (\text{III.52})$$

Le vecteur charge est donné par :

$$F = \langle F_{x1} \ F_{y1} \ F_{x2} \ F_{y2} \ F_{x3} \ F_{y3} \ F_{x4} \ F_{y4} \rangle^T \quad (\text{III.53})$$

III.3.3 Analyse avec l'élément quadrangle Q4

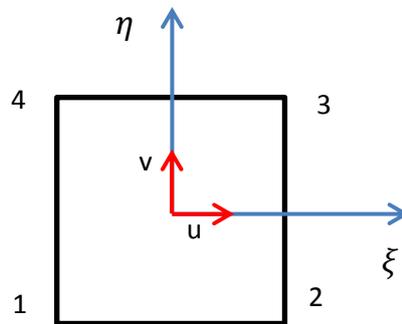
Pour une première approche nous commençons à les modéliser avec des éléments quadrangle de type Q4.

III.3.3.1 Fonctions de forme

Les fonctions de formes d'un élément quadrangle sont :

$$N_1 = \frac{1}{4}(1 - \xi)(1 - \eta) \quad N_2 = \frac{1}{4}(1 + \xi)(1 - \eta) \quad (\text{III.54a})$$

$$N_3 = \frac{1}{4}(1 + \xi)(1 + \eta) \quad N_4 = \frac{1}{4}(1 - \xi)(1 + \eta) \quad (\text{III.54b})$$



III.10 : Elément quadrangle Q4.

III.3.3.2 Matrice Jacobienne et son inverse

La relation (I.32) page 16 du premier chapitre permet de calculer la matrice Jacobienne :

$$[J] = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix} = \begin{bmatrix} N_{1,\xi} & N_{2,\xi} & N_{3,\xi} & N_{4,\xi} \\ N_{1,\eta} & N_{2,\eta} & N_{3,\eta} & N_{4,\eta} \end{bmatrix} \begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ x_3 & y_3 \\ x_4 & y_3 \end{bmatrix} \quad (\text{III.55})$$

$$[J] = \begin{bmatrix} (1 + \eta) & -1(1 + \eta) & -(1 - \eta) & (1 + \eta) \\ (1 + \xi) & (1 - \xi) & -(1 - \xi) & -(1 + \xi) \end{bmatrix} \begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ x_3 & y_3 \\ x_4 & y_3 \end{bmatrix} \quad (\text{III.56})$$

Il est possible de calculer son déterminant : $\det[J] = J_{11}J_{22} - J_{12}J_{21}$.

et son inverse : $[J]^{-1} = \frac{1}{\det(J)} C_J^T$

III.3.3.3 Matrice de rigidité élémentaire [11]

L'adaptation de l'expression (I.9) page 5 du premier chapitre pour la matrice de rigidité élémentaire dans le plan donne :

$$K_e = t \int_{-1}^1 \int_{-1}^1 [B]^T [H] [B] \cdot \det(J) \cdot d\xi d\eta \quad (\text{III.57})$$

$$K_e = \frac{E}{1-\nu^2} t \int_{-1}^1 \int_{-1}^1 [B]^T \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} [B] \cdot \det(J) \cdot d\xi d\eta \quad (\text{III.58})$$

Le champ de déplacement est donné par :

$$\begin{Bmatrix} u \\ v \end{Bmatrix} = \begin{bmatrix} N_1 & 0 & N_2 & 0 & N_3 & 0 & N_4 & 0 \\ 0 & N_1 & 0 & N_2 & 0 & N_3 & 0 & N_4 \end{bmatrix} \begin{Bmatrix} U_1 \\ V_1 \\ U_2 \\ V_2 \\ U_3 \\ V_3 \\ U_4 \\ V_4 \end{Bmatrix} \quad (\text{III.59})$$

Les dérivées des fonctions de forme sont :

$$[B] = [Der][N]$$

$$[B] = \begin{bmatrix} \frac{\partial N_1}{\partial x} & 0 & \frac{\partial N_2}{\partial x} & 0 & \frac{\partial N_3}{\partial x} & 0 & \frac{\partial N_4}{\partial x} & 0 \\ 0 & \frac{\partial N_1}{\partial y} & 0 & \frac{\partial N_2}{\partial y} & 0 & \frac{\partial N_3}{\partial y} & 0 & \frac{\partial N_4}{\partial y} \\ \frac{\partial N_1}{\partial y} & \frac{\partial N_1}{\partial x} & \frac{\partial N_2}{\partial y} & \frac{\partial N_2}{\partial x} & \frac{\partial N_3}{\partial y} & \frac{\partial N_3}{\partial x} & \frac{\partial N_4}{\partial y} & \frac{\partial N_4}{\partial x} \end{bmatrix} \quad (\text{III.60})$$

L'inverse de la matrice Jacobienne nous permet d'écrire les dérivées dans le repère global.

$$[B_{,x}] = [J]^{-1} [B_{,\xi}] \quad (\text{III.61})$$

donc :

$$[B] =$$

$$[J]^{-1} \begin{bmatrix} (1+\eta) & 0 & -1(1+\eta) & 0 & -(1-\eta) & 0 & (1+\eta) & 0 \\ 0 & (1+\xi) & 0 & (1-\xi) & 0 & -(1-\xi) & 0 & -(1+\xi) \\ (1+\xi) & (1+\eta) & (1-\xi) & -1(1+\eta) & -(1-\xi) & -(1+\xi) & - & (1-\eta) \end{bmatrix}$$

(III.62)

Il ne nous reste qu'à remplacer le calcul de l'intégrale de la fonction par une sommation aux points de Gauss-Legendre dont les coordonnées sont : $\xi, \eta = \pm \frac{1}{\sqrt{3}}$ avec leurs poids correspondant $w_i, w_j = [1 \ 1]$.

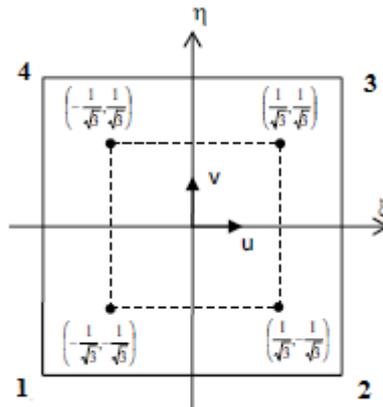


Figure III.11 : Points de Gauss-Legendre.

ce qui nous donne :

$$K_e = t \int_{-1}^1 \int_{-1}^1 [B]^T [H] [B] \cdot \det(J) d\xi d\eta = t \sum_{i=1}^2 \sum_{j=1}^2 [B]^T [H] [B] \cdot \det(J) \cdot w_i w_j$$

(III.63)

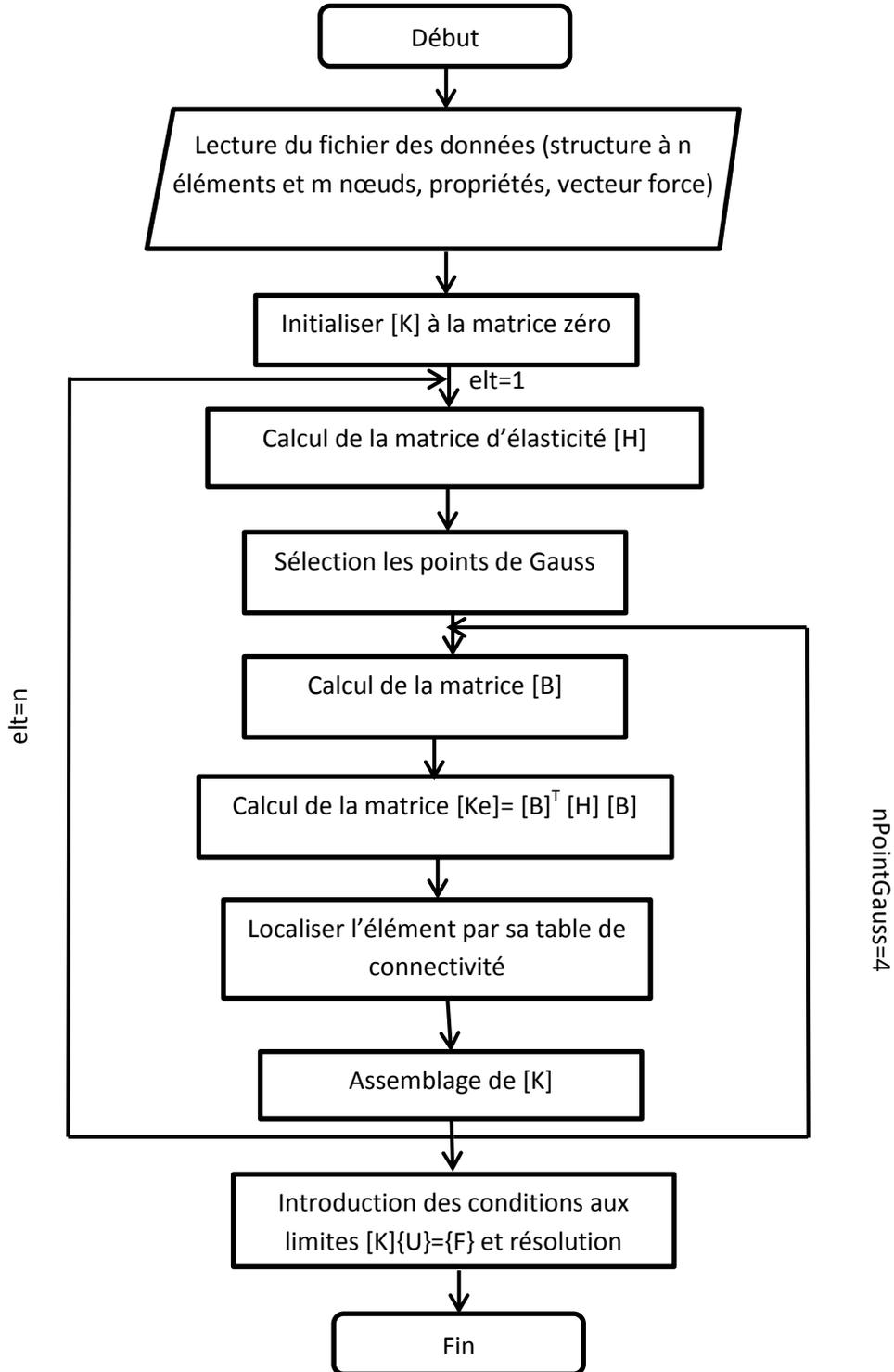
III.3.3.4 Vecteur force élémentaire

La même relation (I.9) permet d'écrire l'expression du vecteur force élémentaire :

$$F_e = \int_{\Gamma} [N]^T \{q\} d\Gamma$$

Seules les faces chargées seront concernées, il est important de dire que ces charges sont appliquées au niveau des nœuds.

III.3.4 Organigramme de résolution



Chapitre IV :
Présentation des
programmes de calcul
développés

Chapitre IV: Présentation des programmes de calcul développés

IV Introduction

La mise en œuvre d'un code de calcul informatique requiert la programmation des opérations suivantes :

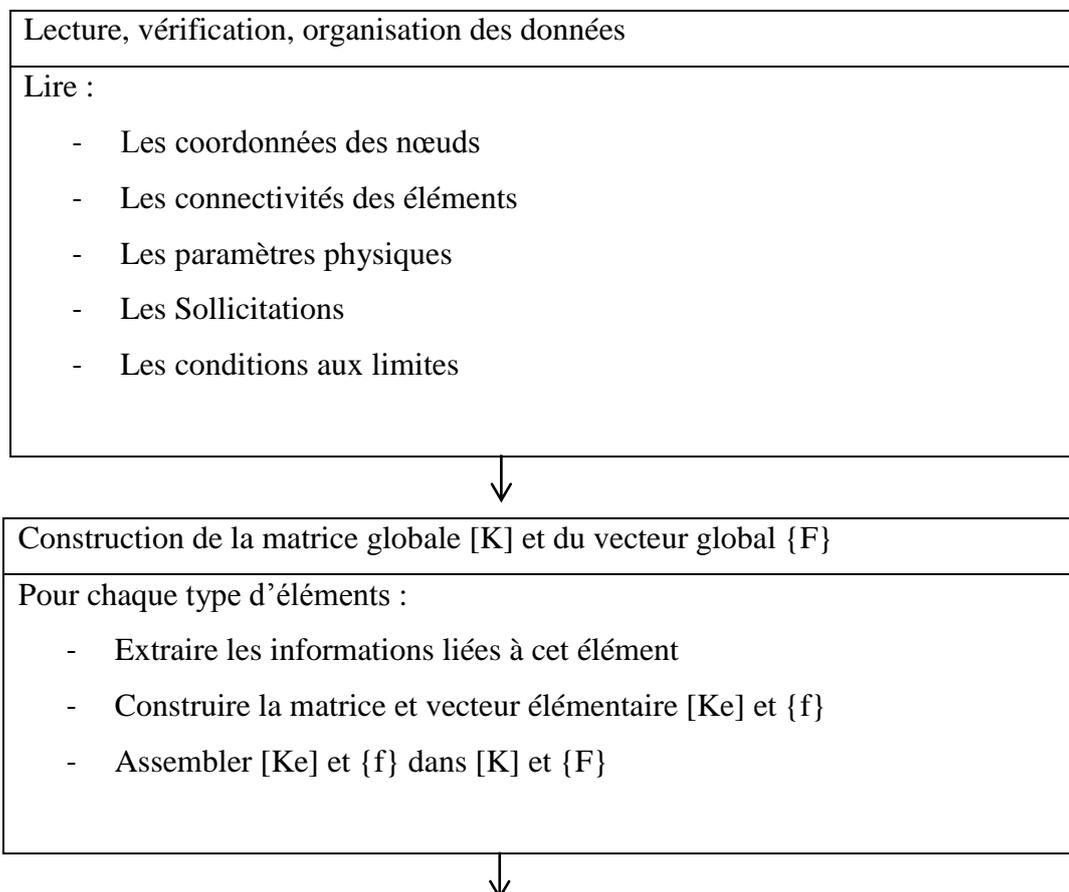
- La description géométrique et physique du problème étudié.
- Le calcul des matrices et vecteurs élémentaires et l'assemblage du système global.
- La résolution du système.
- La visualisation des résultats.

L'utilisation concrète d'un logiciel de modélisation par éléments finis sur le marché donne certains savoir-faire et une expertise adaptée au problème traité, une bonne connaissance de la physique impliquée et une compréhension générale de la méthode des éléments finis.

Nous présentons ici les étapes et les démarches à suivre pour l'élaboration d'un programme de calcul des éléments du chapitre précédent.

IV.1 Etapes caractéristiques d'un programme d'éléments finis

Pour un calcul linéaire et stationnaire, la figure IV.1 montre l'enchaînement du bloc fonctionnel d'un code de calcul par éléments finis.



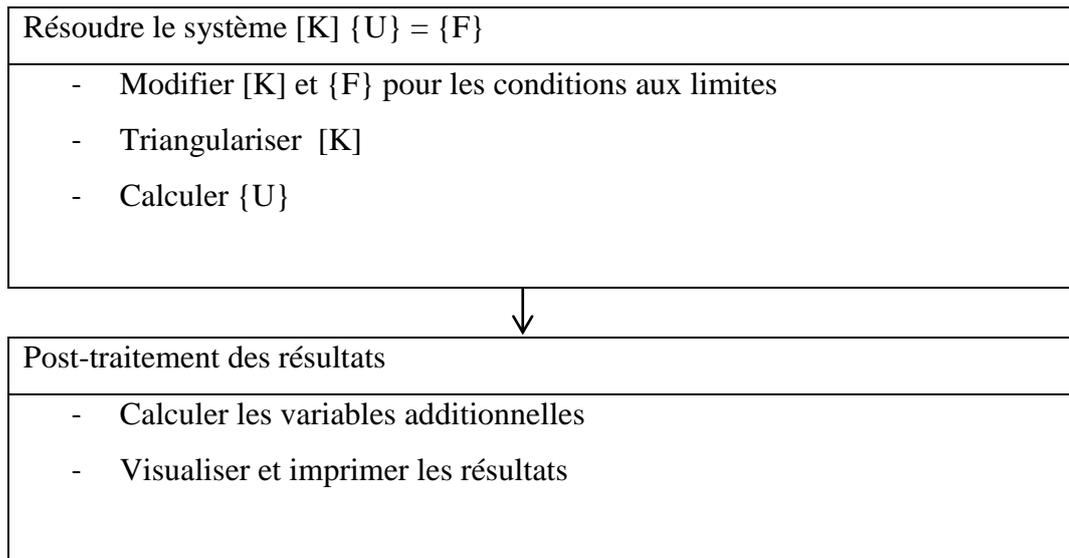


Figure IV.1 : Etapes Caractéristiques d'un programme d'éléments finis.

IV.2 Techniques de programmation

IV.2.1 Possibilité du programme

Un programme de calcul doit être capable de résoudre :

- Des problèmes variés issus de domaines différents : élasticité linéaire ou non linéaire.
- Utilisation des données fournit par les outils de CAO, de génération de maillage automatique et d'adaptation du maillage.
- Problème à une, deux, trois dimensions.
- Nombre de degrés de liberté de chaque nœud.
- Librairie d'éléments finis facile à enrichir.
- Outils de post-traitement pour la visualisation des résultats avec des possibilités d'animation.

IV.2.2 Modularité

Un programme de calcul est forcément volumineux et complexe, il faut le structurer d'une manière modulaire pour que :

- Sa logique soit facile à comprendre.
- Il soit facile à modifier.

Chapitre IV: Présentation des programmes de calcul développés

- Plusieurs personnes puissent collaborer à son développement sans que chacun ait à connaître parfaitement l'ensemble du programme.
- L'on puisse spécialiser ou optimiser le programme pour un type d'application donné par simple remplacement de certains sous-programmes.

IV.2.3 Langage de programmation

Le choix du langage dépend de la manière avec laquelle nous voulons procéder pour concevoir et coder le programme, il existe deux types :

-Merise : pour la programmation procédurale et fonctionnelle c'est notre cas.

-UML : pour la programmation orientée objet.

Pour notre étude le choix est fixé au langage MATLAB qui est un EDI interactif basé sur le calcul matriciel (MATrix LABoratory). Il offre une syntaxe très souple au vu des possibilités de calcul numérique très grandes grâce à une riche bibliothèque de fonctions prêtes à l'emploi (toolbox).

IV.3 Programmes développés

Il existe de nombreux logiciels généraux d'excellente qualité qui permettent de résoudre un ensemble de problèmes physiques couramment rencontrés dans l'industrie, notre objectif n'est pas de concurrencer ces logiciels qui incluent un savoir-faire important et accumulé depuis les années cinquante, mais d'aider l'ingénieur à bien comprendre les étapes de la mise en œuvre de la méthode des éléments finis et proposer une structure simple des normes de programmations permettant de construire rapidement un code de calcul.

IV.3.1 Programme d'élément barre à deux nœuds

IV.3.1.1 Organisation

Tout l'ensemble des fichiers du programme sont répartis dans le (tableau IV.1) :

Fichier	Description
MEFbarre.m	Fichier fonction principal du solveur.
barre2dK.m	Fichier fonction qui calcul la matrice de rigidité globale K après assemblage.

Chapitre IV: Présentation des programmes de calcul développés

Localiser.m	Fichier fonction de localisation de l'élément en cours de traitement.
barre2dKe.m	Fichier fonction qui calcul la matrice de rigidité élémentaire K_e .
EltOrientLong.m	Fichier fonction pour le calcul de la longueur et l'orientation d'un élément.
delDOFs.m	Fichier fonction pour appliquer les conditions aux limites pour réduire le système à partir du vecteur d'encastrement.
fichierdonnee.m	Fichier fonction qui contient les données du problème à résoudre.

Tableau IV.1 : L'ensemble des fichiers du programme MEFbarre.

IV.3.1.2 Fichier définissant le problème à résoudre

Algorithme

- % Table des coordonnées

```
tablcoord = [ x1  y1  
             x2  y2  
             ...  ... ];
```

- % Table des connectivités

```
tablconnec = [ élément1  
             élément2  
             ... ];
```

- % Vecteur d'encastrement

```
encast = [ 1 ; 2 ; ... ] ; % chaque nœud possède deux ddl
```

- % Propriétés des éléments

$A = [\text{ones}(\text{size}(\text{tablconnec},1))] * \text{Aire de la section droite}$; % la longueur du vecteur est égale à la longueur de la table des connectivités.

$E = [\text{ones}(\text{size}(\text{tablconnec},1))] * \text{module de Young}$; % sa longueur est la même que celle de la table des connectivités.

- % Vecteur force appliqué

$F = [\text{val1} ; \text{val2} ; \text{val3} ; \dots]$; % sa taille est égale deux fois la taille de la table des coordonnées , le signe est selon la direction de cette charge.

IV.3.1.3 Les algorithmes des différentes fonctions

a) la fonction MEFbarre

❖ Début

- *Lecture du fichier données (tablcoord, tablconnec, ...etc)*
- *Appel de la fonction « $[K]=\text{barre2dK}(\text{tablcoord},\text{tablconnec},A,E)$ » qui calcul la matrice de rigidité globale*
- *Affectation de K à T ($T=K$) pour calculer les réactions*
- *Appel de la fonction « $M=\text{delDOFs}(M,v)$ » pour les conditions aux limites (conditions d'appuis) de la matrice K et du vecteur F*
- *Résolution $U=K\backslash F$*
- *Calcul des réactions $R=T*U$*
- *Affichages des solutions*

❖ Fin

b) la fonction barre2dK

❖ Début

- *Calcul du nombre d'éléments total de la table des connectivités avec la fonction « $\text{neltotal} = \text{size}(\text{tablconnec},1)$ »*
- *Calcul du nombre de degrés de libertés total du système avec la table des coordonnées avec la fonction « $\text{nddltotal} = 2*\text{size}(\text{tablcoord},1)$ »*
- *Initialisation de la matrice de rigidité global « $K=\text{zeros}(\text{nnoeudtotal},\text{nnoeudtotal})$*
- *Début de boucle sur le nombre des éléments total $i=1 : \text{nelttotal}$*
 - *Repérer l'élément en cours de traitement « $\text{ti}=[\text{tablconnect}(i, :)]$ »*
 - *Calcul des coordonnées de cet élément « $\text{tablcoordXY}=\text{tablcoord}(\text{ti}, :)$ »*
 - *Appel de la fonction « $\text{Ke}=\text{barre2dKe}(\text{tablcoord}(\text{tablcoordXY}, A(i), E(i)))$ » pour calculer la matrice de rigidité élémentaire de l'élément*
 - *appel de la fonction « $\text{Li} = \text{Localiser}(\text{ti})$ » pour localiser l'élément*
 - *Assemblage avec la table de localisation « $K(\text{Li},\text{Li})= K(\text{Li},\text{Li})+\text{Ke}$*
- *Fin de boucle*

❖ Fin

c) la fonction Localiser

❖ Début

- *Calcul de nombre de nœuds par élément « $\text{nnoeudelt}=\text{length}(\text{tablconnec})$ »*
- *Début de la boucle « $i=1 : \text{nnoeudelt}$ »*

- $L([2*i-1 \ 2*i]) = [2*tablconnec(i)-1 \ 2*tablconnec(i)]$

- *Fin de la boucle*

❖ Fin

d) la fonction barre2dKe

❖ Début

- *Appel de la fonction « $[L,c,s]=EltLongOrient(tablcoordXY)$ » pour calculer la longueur de l'élément et les son orientation (cos, sin)*
- *Calcul de la matrice de rotation R*
- *Calcul de $Ke = (A * E / L) * R$*

❖ Fin

e) la fonction EltLongOrient

❖ Début

- *Calcul de la longueur horizontale x de l'élément ($x = x2-x1$)*
- *Calcul de sa longueur verticale y ($y = y2-y1$)*
- *Calcul de sa longueur diagonale de l'élément « $L = \text{sqrt}(x*x + y*y)$ »*
- *Calcul du cosinus « $c = x/L$ »*
- *Calcul du sinus « $s = y/L$ »*

❖ Fin

e) la fonction delDOFs

❖ Début

- *Tri du vecteur d'encastrement*
- *Calcul de sa longueur « $n = \text{length}(v)$ »*
- *Test si la longueur de la matrice d'entrée est égal à 1 (vecteur)*
 - *Début de la boucle « $i = n : -1 : 1$ »*
 - *affecter la valeur zéro à la ligne du vecteur en cours « $M(v(i)) = 0$ »*
 - *Fin de la boucle*
 - *Sinon début de la boucle « $i = n : -1 : 1$ » (matricie)*
 - *affecter la valeur zéro à la ligne de la matrice en cours « $M(v(i), :) = 0$ »*
 - *affecter la valeur zéro à la colonne de la matrice en cours « $M(:, v(i)) = 0$ »*
 - *affecter la valeur 1 à la diagonale « $M(v(i), v(i)) = 1$ »*
- *Fin de la boucle*

Chapitre IV: Présentation des programmes de calcul développés

- *Fin du test*
- ❖ Fin

IV.3.2 Programme d'élément poutre à deux nœuds

IV.3.2.1 Organisation

Les fichiers du programme sont répartis dans le (tableau IV.2) :

Fichier	Description
MEFpoutre.m	Fichier fonction principal du solveur.
poutre2dK.m	Fichier fonction qui calcul la matrice de rigidité globale K.
poutre2dKe.m	Fichier fonction qui calcul la matrice de rigidité élémentaire K_e .
EltOrientLong.m	Fichier fonction pour le calcul de la longueur et l'orientation d'un élément.
Assembler	Fichier fonction pour assembler la matrice de rigidité globale K.
poutreChargeRepartie.m	Fichier fonction qui calcul la charge répartie F_{rep} tout au long de la poutre après assemblage.
poutreFe.m	Fichier fonction qui transforme cette charge répartie en charge nodale.
Localiser.m	Fichier fonction de localisation de l'élément en cour de traitement.
PoutreChageNodale	Fichier fonction pour appliquer l'effort normal (traction/ compression), l'effort vertical et un moment.
delDOFs.m	Fichier fonction pour appliquer les conditions aux limites pour réduire le système à partir du vecteur d'encastrement.
fichierdonnee.m	Fichier fonction qui contient les données du problème à résoudre.

Tableau IV.2 : L'ensemble des fichiers du programme MEFpoutre.

Chapitre IV: Présentation des programmes de calcul développés

IV.3.2.2 Fichier définissant le problème à résoudre

Algorithme

```
- % Table des coordonnées

tablcoord = [ x1  y1
              x2  y2
              ...  ... ];

- % Table des connectivités

tablconnec = [ élément1
              élément2
              ... ];

- % Vecteur d'encastrement
encast = [ 1 ; 2 ; ... ]; % chaque nœud possède trois ddl

- % Propriétés des éléments

b=[ones(size(tablconnec,1))]*val ; % la base de la section droite

h= [ones(size(tablconnec,1))]*val ; % la hauteur de la section droite

E= [ones(size(tablconnec,1))]*module de Young ; % module de Young

- % Vecteur force nodal appliqué

f=[numéro_du_nœud ,val1 ,val2 ,val3
  ... ,val1, val2, val3      ]; % ou le premier paramètre désigne le numéro
du nœud ou on veut appliquer la charge, val1 est l'effort normal, val2 l'effort vertical, val3 le
moment.

- % vecteur charge répartie

q= [numéro_de_l'élément, q0,qL
  ... , q0, qL] ; % le premier paramètre désigne le numéro de l'élément
ou appliquer la charge répartie, q0 et qL sont injectés dans la fonction poutreFe pour pouvoir
remplacer la charge répartie par une autre concentré sur les nœuds.
```

IV.3.2.3 Les algorithmes des différentes fonctions

a) la fonction MEFpoutre

❖ Début

- *Lecture du fichier données (tablcoord, tablconnec, ...etc)*
- *Calcul de nombre de nœuds total « nnoeudstotal=size(tablcoord,1) »*
- *Appel de la fonction « [K]=poutre2dK(tablcoord,tablconnec,b,h,E) » qui calcul la matrice de rigidité globale*
- *Affectation de K à T (T=K) pour calculer les réactions*
- *Appel de la fonction « F=poutreChargeRepartie(tablcoord,tablconnec,q) » pour transformer la charge répartie en charge nodale*
- *Affectation de F à Fq « Fq=F » pour le soustraire du vecteur des réactions*
- *Appel de la fonction « F= poutreChargeNodale(f,nnoeudstotal) »*
- *Appel de la fonction « M=delDOFs(M,v) » pour les conditions aux limites (conditions d'appuis) de la matrice K et du vecteur F*
- *Résolution $U=K \setminus F$*
- *Calcul des réactions $R=T*U-Fq$*
- *Affichages des solutions*

❖ Fin

b) la fonction poutre2dK

❖ Début

- *Calcul du nombre d'éléments total de la table des connectivités avec la fonction « neltotal = size(tablconnec ,1) »*
- *Calcul du nombre de degrés de libertés total du système avec la table des coordonnées avec la fonction « nddltotal = 3*size(tablcoord ,1) »*
- *Initialisation de la matrice de rigidité globale « K=zeros(nddltotal, nddltotal)*
- *Début de boucle sur le nombre des éléments total « i=1 : neltotal »*
 - *Repérer l'élément en cours de traitement « ti=[tablconnect(i, :)] »*
 - *Calcul de ses coordonnées « tablcoordXY=tablcoord(ti, :) »*
 - *appel de la fonction « Ke=barre2dKe(tablcoordXY, A(i), E(i)) » pour calculer la matrice de rigidité élémentaire de l'élément*
 - *Calcul « Li = Localiser (ti, :) » pour localiser l'élément*
 - *Appel de la fonction « K=assembler(K,Ke,Li) » pour assemblage avec la table de localisation « K(Li,Li)= K(Li,Li)+Ke*
- *Fin de boucle*

❖ Fin

c) la fonction barre2dKe

❖ Début

- Appel de la fonction « $[L,c,s]=EltLongOrient(tablcoordXY)$ » pour calculer la longueur de l'élément et les son orientation (cos, sin)
- Calcul de la surface « $A=b*h$ » et le moment d'inertie « $I= (b*h^3)/12$ »
- Calcul de la matrice de rotation R
- Calcul de $Ke1$
- Calcul de $Ke=R^T*Ke1*R$

❖ Fin

d) la fonction Localiser

❖ Début

- Initialiser une matrice identité « $e=eye(n)$ »
- Appel de la fonction « $L=kron(n*tablconnec,e(n, :))$ » pour calculer le produit tensoriel
- Début de la boucle « $i=1 :n-1$ »
 - sommation du vecteur « $L =L + kron(n*tablconnec-i,e(n-i, :))$ »
- Fin de la boucle

❖ Fin

e) la fonction EltLongOrient

❖ Début

- Calcul de longueur horizontale x de l'élément ($x = x2-x1$)
- Calcul de sa longueur verticale y ($y= y2-y1$)
- Calcul de la longueur diagonale de l'élément « $L=sqrt(x*x+y*y)$ »
- Le cosinus « $c=x/L$ »
- Le sinus « $s=y/L$ »

❖ Fin

f) la fonction poutreChargeRepartie

❖ Début

- Calcul du nombre d'éléments à charger « $neltcharge=size(q,1)$ »
- Calcul du nombre de degré de liberté total
« $nnoeudtotal=3*size(tablconnec,1)$ »
- Initialiser F à zéro « $F=zeros(nnoeudtotal)$ »
- Début de la boucle « $i=1 :neltcharge$ »
 - Repérer l'élément « $ie= q(i,1)$ »
 - Repérer la première charge $q0$ « $q0=q(i,2)$ »
 - Repérer la deuxième charge qL « $qL=q(i,3)$ »

Chapitre IV: Présentation des programmes de calcul développés

- Repérer la table de connectivité de l'élément qu'on veut charger
« $ti=tblconnec(ie, :)$ »
- Repérer la table des coordonnées « $tblcoordXY=tblcoord(ti, :)$ »
- Appel de la fonction « $Fe=poutreFe(tblcoordXY,q0,qL)$ » pour calculer la charge répartie
- Appel de la fonction « $L=Localiser(ti,3)$ » pour localiser l'élément
- Assemblage « $F(L)=F(L)+Fe$ »

- Fin de la boucle

❖ Fin

g) la fonction poutreFe

❖ Début

- Appel de la fonction « $L=ElongOrient(tblcoordXY)$ »
- Calcul de Fe

❖ Fin

h) la fonction poutreChargeNodale

❖ Début

- Calcul du nombre de nœuds à charger « $nnoeudcharge=size(f,1)$ »
- Initialiser le vecteur F à zéro « $F=zeros(3*nnoeudcharge,1)$ »
- Début de la boucle « $i=1:nnoeudcharge$ »
 - Repérage du nœud à charger « $nccharge=f(i,1)$ »
 - appel de la fonction « $L=Localiser(nccharge,3)$ »
 - assemblage « $F(L)=[f(i,2) f(i,3) f(i,4)]$ »
- Fin de la boucle

❖ Fin

i) la fonction delDOFs

❖ Début

- Tri du vecteur d'encastrement
- Calcul de sa longueur « $n=length(v)$ »
- Test si la longueur de la matrice d'entrée est égal à 1 (vecteur)
 - Début de la boucle « $i=n:-1:1$ »
 - affecter la valeur zéro à la ligne du vecteur en cours « $M(v(i))=0$ »
 - Fin de la boucle
 - Sinon début de la boucle « $i=n:-1:1$ » (matricie)
 - affecter la valeur zéro à la ligne de la matrice en cours « $M(v(i),:)=0$ »
 - affecter la valeur zéro à la colonne de la matrice en cours
« $M(:,v(i))=0$ »
 - affecter la valeur 1 la diagonale « $M(v(i),v(i))=1$ »
- Fin de la boucle

- *Fin du test*
- ❖ Fin

IV.3.3 Programme d'élément membrane

IV.3.3.1 Organisation

L'ensemble des fichiers du programme sont répartis dans le (tableau IV.3) :

Fichier	Description
MEFmembrane.m	Fichier script principal du solveur.
Localiser.m	Fichier fonction de localisation de l'élément en cours de traitement.
elt_q4_ke.m	Fichier fonction qui calcul la matrice de rigidité élémentaire K_e .
delDOFs.m	Fichier fonction pour appliquer les conditions aux limites pour réduire le système à partir du vecteur d'encastrement.
Elt_q4_gr.m	Fichier fonction qui calcul les contraintes.
fichierdonnee.m	Fichier fonction qui contient les données du problème à résoudre.

Tableau IV.3 : L'ensemble des fichiers du programme MEFmembrane.

IV.3.3.2 Fichier définissant le problème à résoudre

Algorithme :

- % nombre de degré de liberté par noeud
 ndlnoeud = 2 ;
- % Table des coordonnées
 tablcoord = [x1 y1
 x2 y2
 ] ;
- % Table des connectivités
 tablconnec = [élément1
 élément2
 ...] ;

Chapitre IV: Présentation des programmes de calcul développés

- % Calcul de différents paramètres

```
nnoeudtotal=size(tablcoord,1); % nombre de nœuds total  
nnoeudel=size(tablconnec,2); % nombre de nœud par élément  
neltotal=size(tablconnec,1); % nombre de nœud total  
ndltotal=ndlnoeud*nnoeudtotal; % nombre de degré de liberté total
```
- % vecteur des propriétés global des éléments

```
preltglobal=[1 210e9 0 0 0.1 0 0]; % première valeur est rho la masse  
volumique, deuxième module de Young E, troisième coefficient de Poisson nu, quatrième  
paramètre de contrainte/déformation plane égal à zéro, cinquième est l'épaisseur, charge sur tous  
les nœuds suivant x, la dernière suivant y.
```
- % Propriété de chaque élément

```
propelt = [1];
```
- % Vecteur d'encastrement on l'initialise à zéro puis on repère les nœuds a
encastrer on les affecte la valeur de 1

```
noeudencast= zeros(1,ndltotal);  
noeudencast([noeud1 noeud2 ....]) = 1;
```
- % puis on applique les conditions de Dirrechlet (les déplacement imposés)

```
encast = zeros(1, ndltotal);  
encast([ noeud1 noeud2 ...]) = [val1 val2 val3 .....];
```
- % Vecteur sollicitation

```
Fc=[ val1 val2 val3 ....] % sa taille est égal au nombre de degré de liberté total du  
système.
```

IV.3.3.3 Les algorithmes des différentes fonctions

a) la fonction MEFmembrane

❖ Début

- *Lecture du fichier données (tablcoord, tablconnec, ...etc)*
- *Initialiser la viable « ndlelt =8 »*
- *Initialiser la matrice de rigidité globale « Kg=zeros(ndltotal) »*
- *Affectation du vecteur sollicitation à Fg « Fg=Fc »*
- *Début de la boucle « ie=1 :neltotal »*
 - *Localiser l'élément en cours de traitement*
« *L=Localiser(ie,nnoeudel,tablconnec,ndlnoeud) »*
 - *repérer les propriétés de l'élément en cours*
« *vprelt=preltglobal(propelt(ie), :)* »

- Appel de la fonction
« $[Ke, Fe] = \text{elt_q4_ke}(ie, \text{tablcoord}, \text{tablconnec}, \text{vprelt}, \text{ndlelt})$ » pour calculer la matrice de rigidité et le vecteur force élémentaire
- assemblage de la matrice de rigidité globale avec la table de localisation
« $Kg(L, L) = Kg(L, L) + Ke$ »
- assemblage du vecteur de force global avec la table de localisation
« $Fg(L, 1) = Fg(L, 1) + Fe$ »
- Fin de la boucle
- Application des conditions aux limites
« $[Kg, Fg] = \text{delDOFs}(\text{noeudencast}, \text{encast}, \text{ndltotal}, Kg, Fg)$ »
- Résolution $U = Kg \setminus Fg$
- Affichages des solutions
- Début de la boucle « $i=1 : \text{neltotal}$ »
 - Localiser le vecteur solution
« $Li = \text{Localiser}(ie, \text{nnoeudel}, \text{tablconnec}, \text{ndlneoud})$ »
 - affectation « $Ue = U(Li)$ »
 - Appel de la fonction « $\text{vsig} = \text{elt_q4_gr}(ie, \text{tablcoord}, \text{tablconnec}, \text{vprelt}, Ue)$ » pour calculer les contraintes
- Fin de la boucle

❖ Fin

b) la fonction Localiser

❖ Début

- Initialiser « $L = []$ » à un vecteur vide
- Début de la boucle « $i=1 : \text{nnoeudelt}$ »
 - test si les éléments de la table de connectivité sont supérieurs à zéro « $\text{tablconnec}(ie, i) > 0$ »
 - Soustraire 1 de l'élément courant de la table de connectivité et l'affectant à la variable « $ln = L(ie, i) - 1$ »
 - Localisation de chaque degré de liberté des nœuds
« $L = [L, (ln) * \text{ndln} + [1 : \text{ndln}]]$ »
 - Fin du test
- Fin de la boucle

❖ Fin

c) la fonction elt_qt_ke

❖ Début

- Calcul des points de Gauss
- Calcul de la matrice d'élasticité
- Initialisation de Ke et Fe à zéro
- Début de la boucle « $ipf=1 : \text{npointGauss}$ »

Chapitre IV: Présentation des programmes de calcul développés

- calcul de N et dN
- calcul de J et $\det J$ et son inverse
- construction de la matrice B
- calcul de la matrice « $K_e = K_e + B' * H * B$ »
- calcul de « $F_e = F_e + N' * F$ »

- Fin de la boucle

❖ Fin

c) la fonction delDOFs

❖ Début

- Début de la boucle « $i = \text{Indltotal}$ » pour parcourir tous les nœuds
 - test les nœuds qui ont la valeur 1
 - soustraire du vecteur force la ligne courante de la matrice de rigidité la valeur courante du vecteur d'encastrement « $F_g = F_g - K_g(:, i) * \text{encast}(i)$ »
 - affecter zéro a toute les composantes de la ligne courante « $K_g(i, :) = 0$ »
 - affecter zéro a toute les composantes de la colonne courante « $K_g(:, i) = 0$ »
 - affecte a la composante courante du vecteur force la valeur de la composante courante du vecteur d'encastrement « $F_g(i) = \text{encast}(i)$ »
 - Fin du test
- Fin de la boucle

❖ Fin

c) la fonction elt_qt_ke

❖ Début

- Calcul des points de Gauss
- Calcul de la matrice d'élasticité
- Initialisation de K_e et F_e à zéro
- Début de la boucle « $\text{ipf} = 1 : \text{npointGauss}$ »
 - calcul de N et dN
 - calcul de J et $\det J$ et son inverse
 - construction de la matrice B
 - calcul des contraintes « $\text{Sig}(i, :) = (H * B * U_e)'$ »
- Fin de la boucle

❖ Fin

Chapitre IV :

Application

Chapitre V : Applications

Une fois que les programmes sont mis en œuvres, il ne nous reste qu'à faire une série de tests pour chaque élément de structure développé auparavant, les résultats obtenus par ses programmes seront comparés à ceux du logiciel ANSYS. Pour les valider, nous avons regroupé cette série de tests en trois parties :

- partie 1 : elle est consacrée à des assemblages d'éléments de barre, nous testons le programme par deux exemples. Le premier fait l'objet d'un treillis de sept barres, les charges sont appliquées dans deux directions différentes. Le deuxième exemple est aussi une structure en treillis qui contient onze barres chargé dans la même direction.

- partie 2 : la deuxième partie concerne l'assemblage des poutres, le premier exemple représente une poutre encastree, chargé verticalement par une force ponctuelle en plus d'un moment par rapport à l'axe z. Pour le deuxième exemple il s'agit d'un portique, on lui applique une force horizontale et un moment.

- partie 3 : la troisième partie s'effectue sur l'élément membrane, nous commençons par mailler une plaque carrée avec un seul élément puis on raffine le maillage pour voir son effet sur les résultats, on termine avec la même démarche pour une plaque rectangulaire trouée.

Les éléments utilisés dans ANSYS pour la comparaison des résultats sont regroupés dans le tableau suivant :

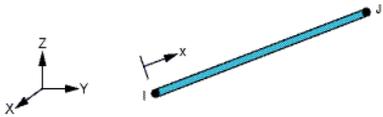
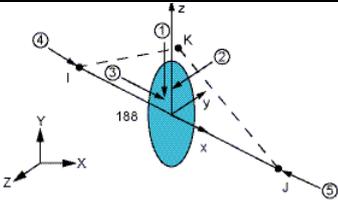
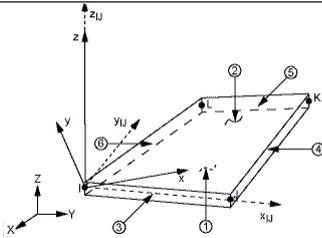
Type d'élément	Géométrie	Description
LINK 180		Elément barre à trois degré de liberté par nœud.
BEAM 188		Elément poutre à six degré de liberté par nœud
SHELL 41		Elément membrane à trois degré de liberté par nœud

Tableau V.1 : Liste des éléments utilisés dans ANSYS.

Chapitre V : Applications

V.1 Application : élément de barre

V.1.1 Exemple 1 (treillis à sept barres)

Le premier exemple est une structure en treillis à une dimension, composée de sept (7) barres sur un appui double au niveau du nœud 1 et un appui simple sur le nœud 5, la charge au nœud 3 est $F_y = -300\text{KN}$, pour le nœud 4 $F_x = 200\text{KN}$, voir la (figure V.1).

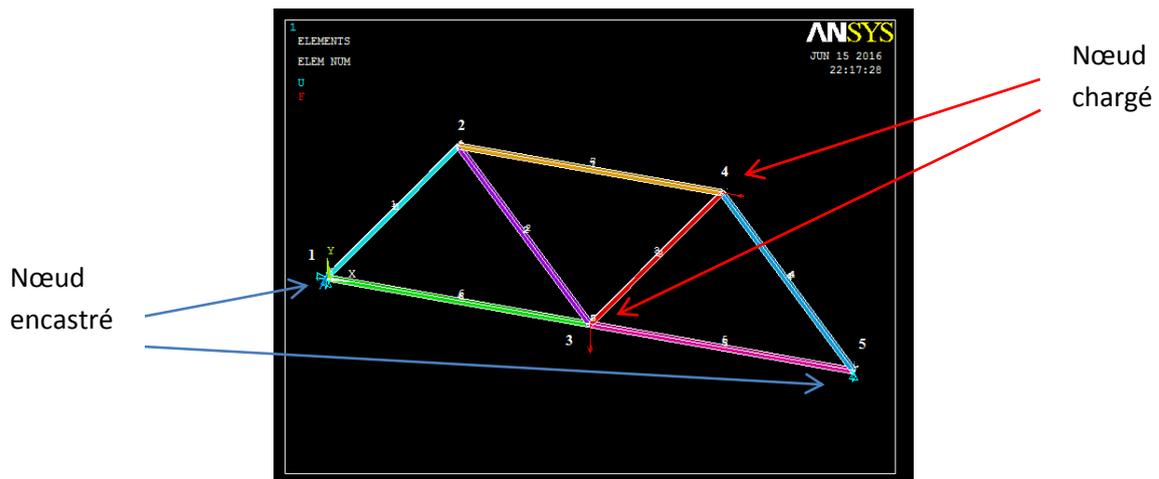


Figure V.1 : Modélisation du treillis à sept barres avec ANSYS.

Les données sont :

- Table des coordonnées de la structure est $[0 \ 0 ; 1500 \ 2000 ; 3000 \ 0 ; 4500 \ 2000 ; 6000]$
- Table des connectivités de la structure $[1 \ 2 ; 2 \ 3 ; 3 \ 1 ; 2 \ 4 ; 4 \ 3 ; 3 \ 5 ; 5 \ 4]$
- Module d'Young de, $E=210e3 \text{ MPa}$.
- Coefficient de poisson, $\nu = 0.3$.
- L'aire de la section droite carrée $A=2500 \text{ mm}^2$.

Il s'agit d'évaluer les déplacements aux nœuds libres de la structure, les résultats obtenus par notre programme MATLAB ainsi que ceux de ANSYS sont présentés dans le (tableau V.2). La figure V.2 nous montre ces déplacements.

Nœuds	Résultats							
	MEFbarre (MATLAB)				ANSYS			
	Déplacements (mm)		Réactions (N)		Déplacements (mm)		Réactions (N)	
	Ux	Uy	Rx	Ry	Ux	Uy	Rx	Ry
1	0.00	0.00	-200000	83333.33	0.00	0.00	-200000	83333.33
2	2.37	-2.40			2.37	-2.40		
3	1.50	-3.67			1.50	-3.67		
4	1.66	-2.18			1.66	-2.18		
5	2.42	0.00	0.00	216666.66	2.42	0.00	0	216666.66

Tableau V.2 : Résultats des déplacements (mm) et des réactions (N) du treillis à sept barres.

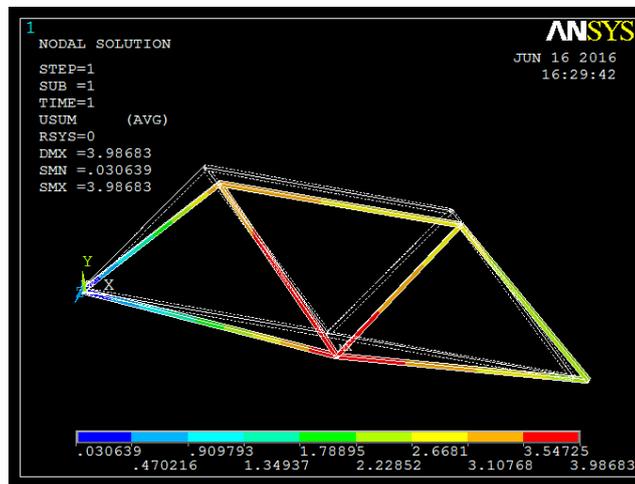


Figure V.2 : Déplacements (mm) de la structure à sept barres avec ANSYS.

Comme nous le voyons dans le tableau ci-dessus les résultats sont identiques, cela revient aux choix des fonctions des formes pour établir la matrice de rigidité élémentaire de l'élément barre.

V.1.2 Exemple 2 (treillis à onze barres)

Pour ce deuxième exemple un autre treillis comportant onze (11) barres sur un appui double au niveau du nœud 1 et un appui simple sur le nœud 5, on applique deux charges identiques $F_y=50\text{KN}$ aux nœuds 3 et 6, une autre charge $F_y=100\text{KN}$ au nœud 4 voir la (figure V.3).

- Table des coordonnées : [0 0 ;3000 0 ;0 3000 ;3000 3000 ; 6000 0 ;6000 3000]
- Table des connectivités :[1 2 ;2 3 ;1 3 ;2 4 ;4 1 ;4 3 ;2 5 ;5 4 ;6 4 ;6 2 ;5 6]

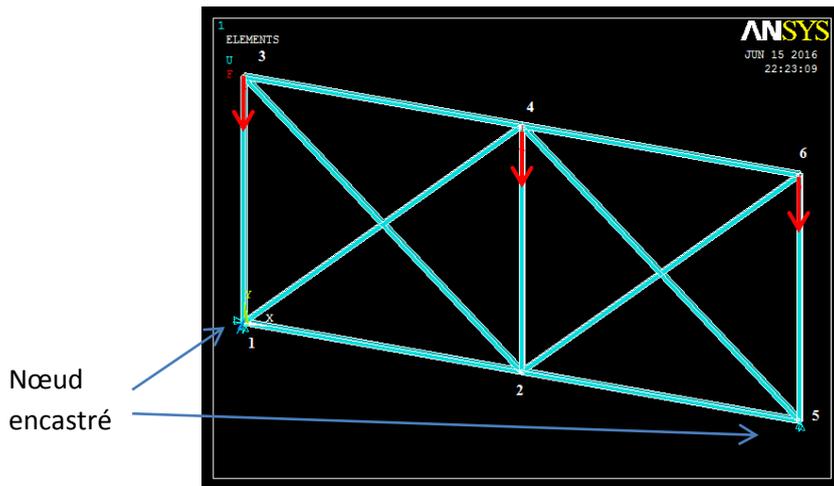


Figure V.3 : Modélisation du treillis à onze barres avec ANSYS.

Les résultats obtenus par le programme MEFbarre et ANSYS sont présentés dans le (tableau V3).

Nœuds	Résultats							
	MEFbarre (MATLAB)				ANSYS			
	Déplacements (mm)		Réactions (N)		Déplacements (mm)		Réactions (N)	
	Ux	Uy	Rx	Ry	Ux	Uy	Rx	Ry
1	0.00	0.00	0.00	100000	0.00	0.00	0.00	100000
2	0.20	-0.65			0.20	-0.65		
3	0.28	-0.36			0.28	-0.36		
4	0.20	-0.80			0.20	-0.80		
5	0.41	0.00	0.00	100000	0.41	0.00	0.00	100000
6	0.13	-0.36			0.13	-0.36		

Tableau V.3 : Résultats des déplacements (mm) et les réactions (N) du treillis à onze barres.

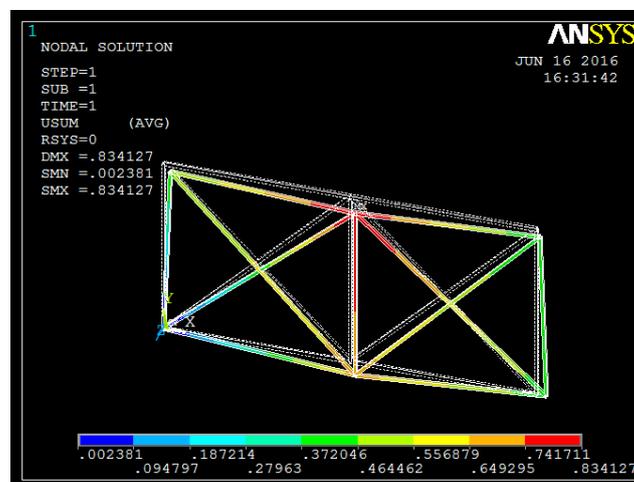


Figure V.4 : Déplacements (mm) de la structure à onze barres avec ANSYS.

Chapitre V : Applications

V.1.3 Interprétation des résultats

L'élément barre que nous avons formulé donne de bon résultats, et se comporte bien quel que soit l'assemblage réalisé.

V.2 Application : élément poutre

V.2.1 Exemple 1 (poutre encastrée)

Ce premier exemple simule le comportement d'une poutre encastrée au nœud 1, une charge ponctuelle $F_y=90$ kN et un moment $M_z=60$ kNm au nœud 2 sont appliquée. (figure V.5).

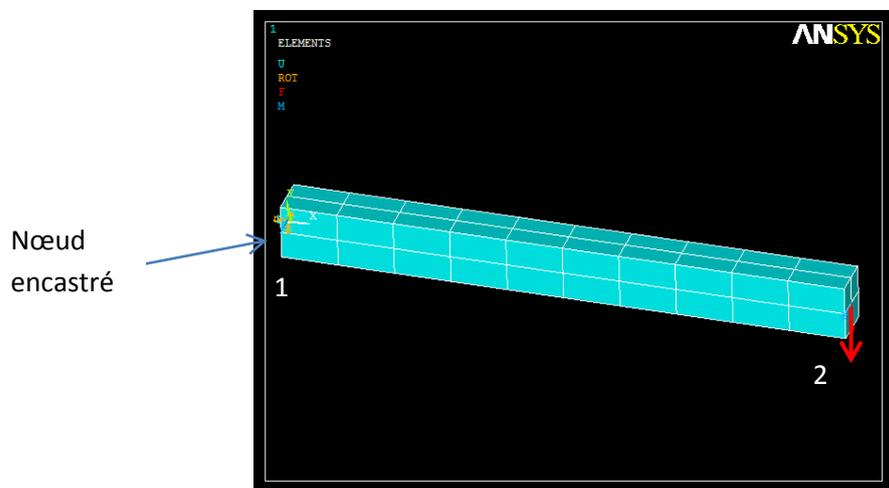


Figure V.5 : Modélisation de la poutre encastrée avec ANSYS.

Les données sont :

- Module d'Young de, $E=210e3$ MPa
- Coefficient de Poisson, $\nu = 0.3$
- Hauteur $h=400$ mm, la base $b=300$ mm
- Table des coordonnées : $[0 \ 0 \ ;4000 \ 0]$
- Table des connectivités : $[1 \ 2]$

Nous avons regroupé l'ensemble des résultats du programme MEFpoutre ainsi que ceux de ANSYS dans le (tableau V.4).

Résultats						
Nœuds	MEFpoutre (MATLAB)			ANSYS		
	Déplacements (mm)		Rotation (mm)	Déplacements (mm)		Rotation (mm)
	Ux	Uy	theta	Ux	Uy	theta
1	0.00	0.00	0.00	0.00	0.00	0.00
2	0.00	-5.71	-2.14E03	0.00	-5.76	-2.14E03

Tableau V.4 : Résultats des déplacements et des rotations (mm) de la poutre encastree.

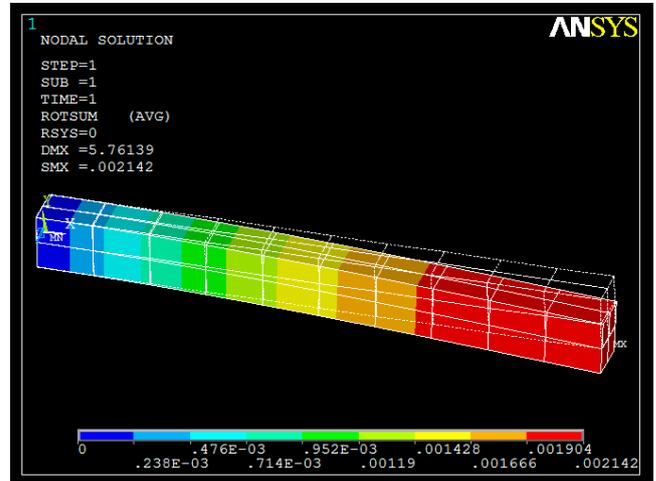
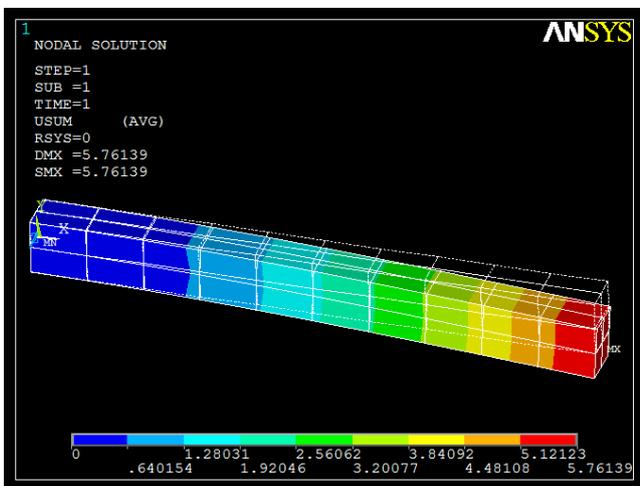


Figure V.6 : Déplacements et rotations (mm) de la poutre encastree avec ANSYS.

V.2.2 Exemple 2 (portique)

L'exemple 2 est une simulation d'un portique soumis à un effort normal $F_x=15\text{KN}$ et un moment $M_z=10\text{KNm}$ au nœud 2, le portique étant encastree aux deux nœuds 1 et 3.

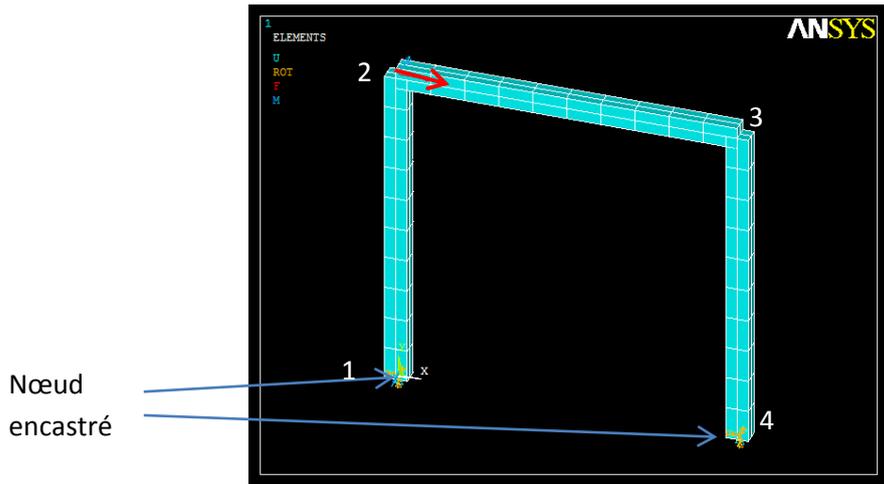


Figure V.7 : Modélisation du portique avec ANSYS.

Avec les mêmes caractéristiques que l'exemple 1, la table des coordonnées ainsi que de la table des connectivités sont données par :

- la table des coordonnées : [0 0 ; 0 6000 ; 6000 6000 ; 6000 0]

- la table des connectivités : [1 2 ; 3 2 ; 3 4]

Résultats						
Nœuds	MEFpoutre (MATLAB)			ANSYS		
	Déplacements (mm)		Rotation (mm)	Déplacements (mm)		Rotation (mm)
	Ux	Uy	theta	Ux	Uy	theta
1	0.00	0.00	0.00	0.00	0.00	0.00
2	0.57	1.53E-03	-6.0E-05	0.57	1.52E-03	-5.8E-05
3	0.57	-1.53E-03	-6.0E-05	0.57	1.52E-03	-5.7E-05
4	0.00	0.00	0.00	0.00	0.00	0.00

Tableau V.5 : Résultats des déplacements et des rotations (mm) du portique.

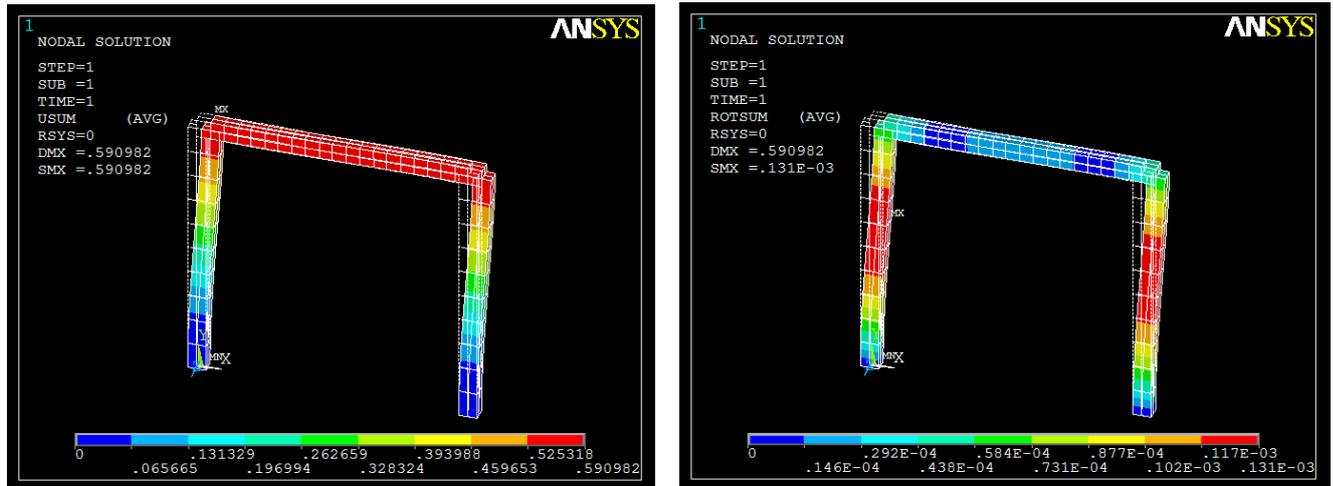


Figure V.8 : Déplacements et rotations (mm) du portique avec ANSYS.

V.2.3 Interprétation des résultats

Les résultats des déplacements des deux programmes indiqués dans les tableaux V.4, V.5 sont très proche, mais pour un maillage différent, dans le cas de ANSYS les résultats des deux exemples sont atteint avec un maillage de 10 éléments, par contre pour le programme MEFpoutre il a suffi d'un seul élément pour aboutir aux même résultats.

V.3 Application : l'élément membrane

V.3.1 Exemple 1 (traction d'une plaque carrée)

Dans cette partie de membrane, il est question de discrétiser la plaque étudiée montrée dans la (figure V.9) à différents maillages, toute en prenant l'effet du coefficient de Poisson qui lie les déformations longitudinales aux déformations transversales. Passer d'un maillage basique à un autre plus raffiné, pour montrer l'effet de la discrétisation géométrique sur la réponse en déplacement de la plaque.

En premier lieu on fixe la valeur du coefficient de Poisson à zéro ($\nu=0$) on applique une charge répartie $q=10^7$ N/m pour différents maillage (1x1), (2x2), (10x10).

La plaque a les dimensions suivantes :

- Coté $a=1000$ mm.
- Epaisseur $t=100$ mm.
- Module de Young du matériau : $E=210$ GPa.

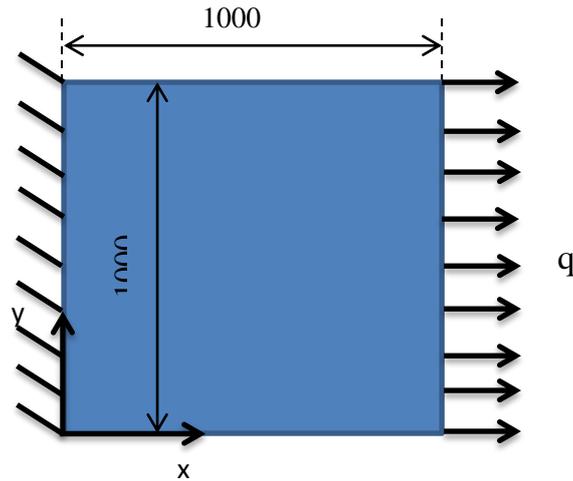


Figure V.9 : Traction de la plaque carrée.

Le tableau V.6 nous donne la valeur maximale des déplacements pour chaque maillage.

Maillage	Résultats			
	MEFmembrane (MATLAB)		ANSYS	
	Déplacements (mm)	Contraintes (MPa)	Déplacements (mm)	Contraintes (MPa)
1x1	0.47	100	0.47	100
2x2	0.47	100	0.47	100
10x10	0.47	100	0.47	100

Tableau V.6 : Résultat du déplacement (mm) et contrainte (MPa) maximal de la plaque pour ($\nu = 0$).



Figure V.10 : Déplacements (mm) et contraintes (MPa) de la plaque pour un maillage (1x1), ($\nu = 0$) avec ANSYS.

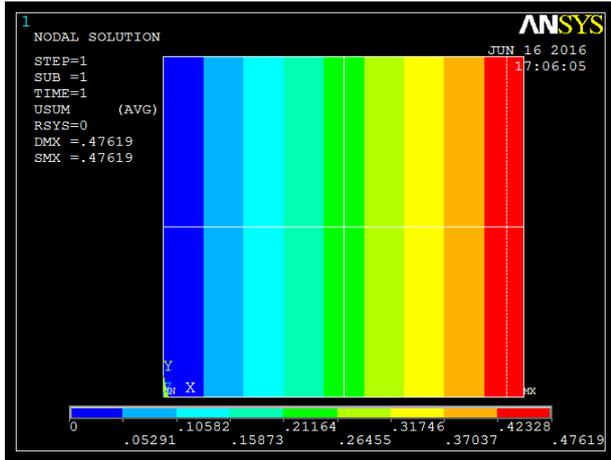


Figure V.11 : Déplacements (mm) et contraintes (MPa) de la plaque pour un maillage (2x2), ($v = 0$) avec ANSYS.

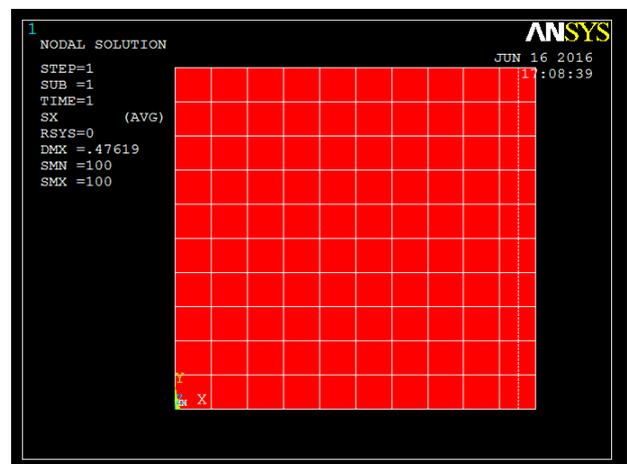
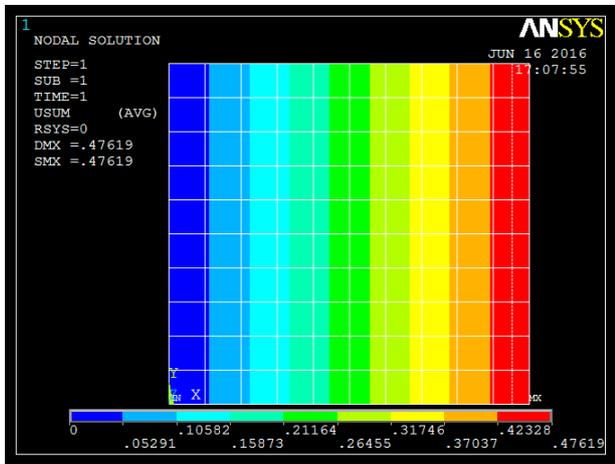


Figure V.12 : Déplacements (mm) et contraintes (MPa) de la plaque pour un maillage (10x10), ($v = 0$) avec ANSYS.

Avec la même démarche qu'auparavant, dans la deuxième partie nous prendrons $v = 0.3$. Nous présentons les résultats du déplacement et contrainte maximale dans le (tableau V.7).

Maillage	Résultats			
	MEFmembrane (MATLAB)		ANSYS	
	Déplacements (mm)	Contraintes (MPa)	Déplacements (mm)	Contraintes (MPa)
1x1	0.47	100	0.46	100
2x2	0.46	106	0.47	104
10x10	0.47	100	0.47	129

Tableau V.7 : Résultat du déplacement (mm) et contrainte (MPa) maximal de la plaque pour $v = 0.3$.

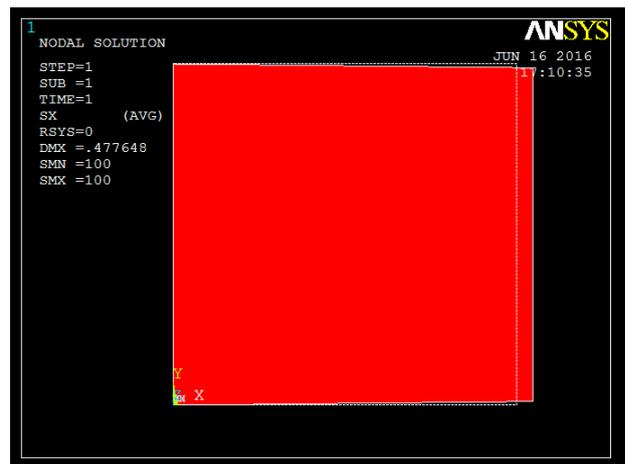
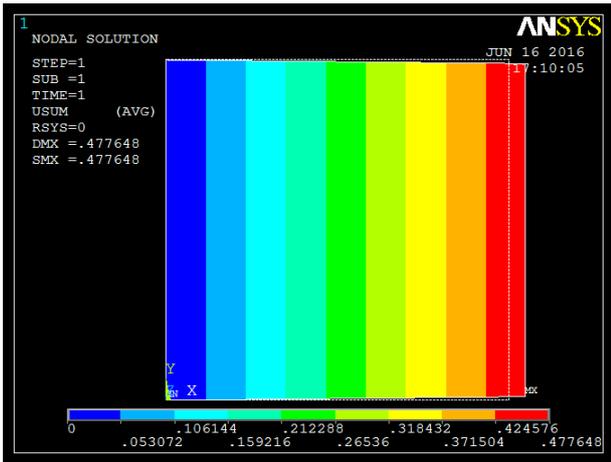


Figure V.13 : Déplacements (mm) et contraintes (MPa) de la plaque pour un maillage (1x1), ($\nu = 0.3$) avec ANSYS.

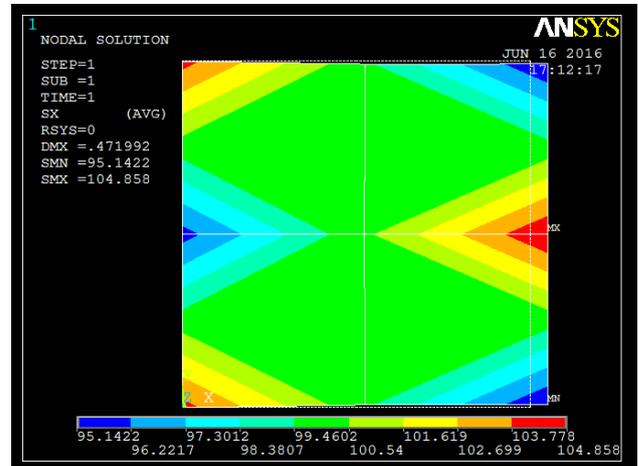
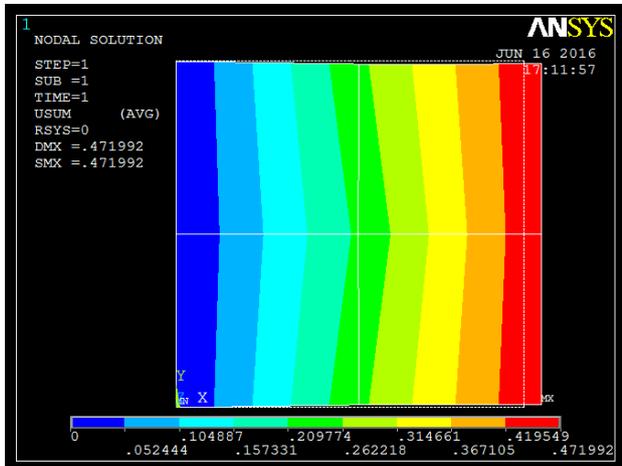


Figure V.14 : Déplacements (mm) et contraintes (MPa) de la plaque pour un maillage (2x2), ($\nu = 0.3$) avec ANSYS.

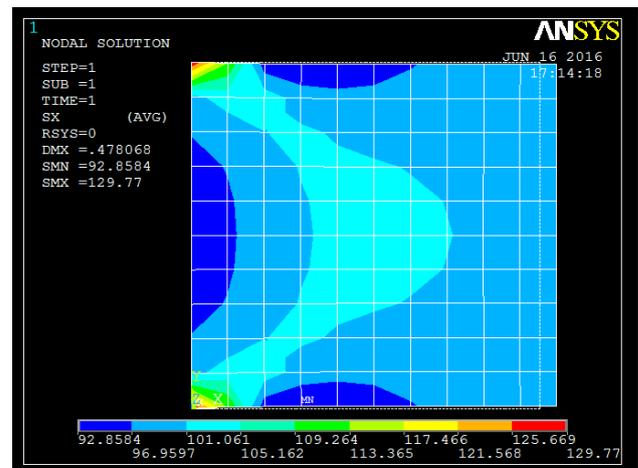
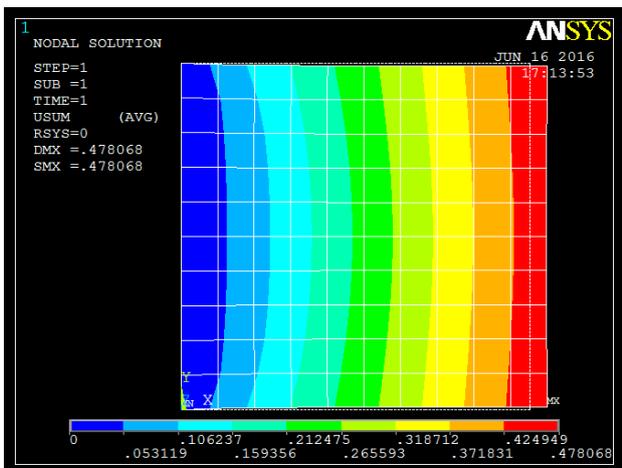


Figure V.15 : Déplacements (mm) et contraintes (MPa) de la plaque pour un maillage (10x10), ($\nu = 0.3$) avec ANSYS.

Chapitre V : Applications

V.3.1.1 Interprétation des résultats

Les résultats numériques obtenus et montrés dans les tableau V.6, V.7 du premier exemple du test de l'élément membrane pour les deux premiers maillages, présentent une satisfaction lorsque $\nu = 0$, mais nous voyons une différence des contraintes du maillage (10x10) quand $\nu = 0.3$. Nous avons constaté pour une bonne application de la charge tout au long de la hauteur de la plaque, il est nécessaire de la répartir sur tous les nœuds. Cette charge est ainsi appelé charge nodale. Les deux nœuds des extrémités de la plaque prennent la moitié de cette charge nodale.

V.3.2 Exemple 2 (traction d'une plaque trouée)

Dans cet exemple notre objectif est de simuler le comportement d'une plaque rectangulaire trouée soumise à une traction et encastré de l'autre coté comme le montre la (figure V.16), les dimensions de cette plaque ainsi que ses propriétés matériau sont :

$L=600$ mm, $h=200$ mm, $t=10$ mm, $r=50$ mm, $E=210$ GPa, $\nu=0.3$, $q=1.10^6$ N/m

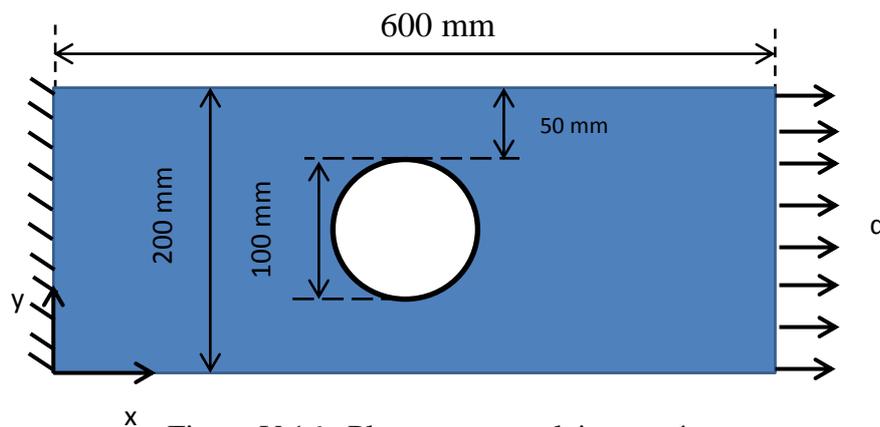


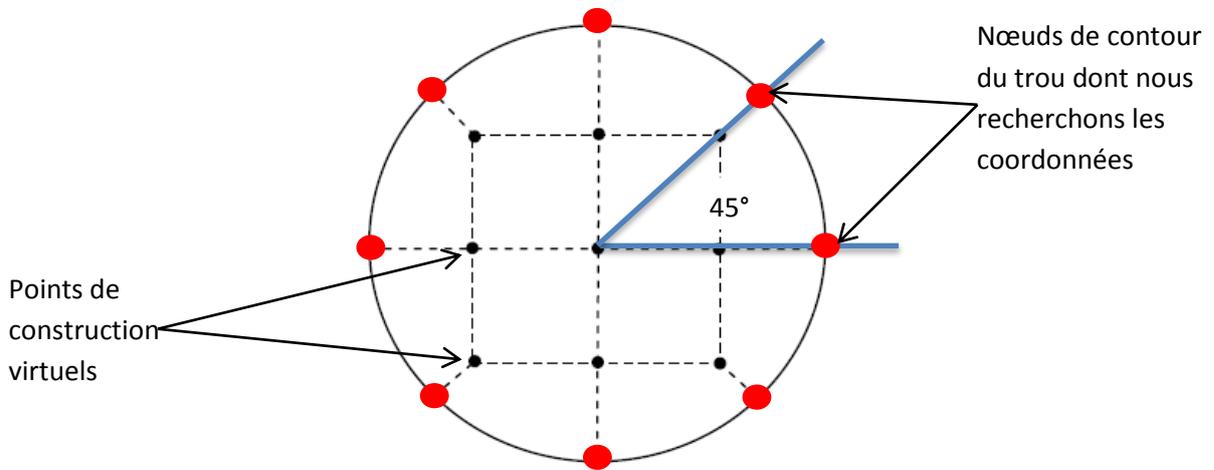
Figure V.16 : Plaque rectangulaire trouée.

Le problème est traité en dimension deux sous l'hypothèse des contraintes planes. Comme vu précédemment, l'approche par éléments finis est basée sur la discrétisation du domaine géométrie, cette approche peut devenir très lourde dans le cas de structures complexes, Les codes de calculs commerciaux possèdent des mailleurs automatiques, pour notre cas on le fait manuellement avec la description de la table des coordonnées et de la table des connectivités.

Certaines formes comme le cercle posent un léger problème, le quadrangle n'étant pas l'idéal pour reconstituer une géométrie curviligne. Nous utilisons alors une technique spécifique (figure V.17) qui consiste à insérer une forme carrée dans le cercle et récupérer les coordonnées des nœuds d'intersection en incrémentant l'angle de 45° pour un premier maillage, puis de même

Chapitre V : Applications

pour les autres angle de $30^\circ, 15^\circ$ du deuxième et troisième maillage, il ne reste qu'à les connectés avec les autres nœuds de la plaque.



V.17 : Maillage d'un cercle avec l'élément Q4 pour un angle de 45° .

Ce qui nous donne pour les trois maillages illustré dans la (figure V.18) :

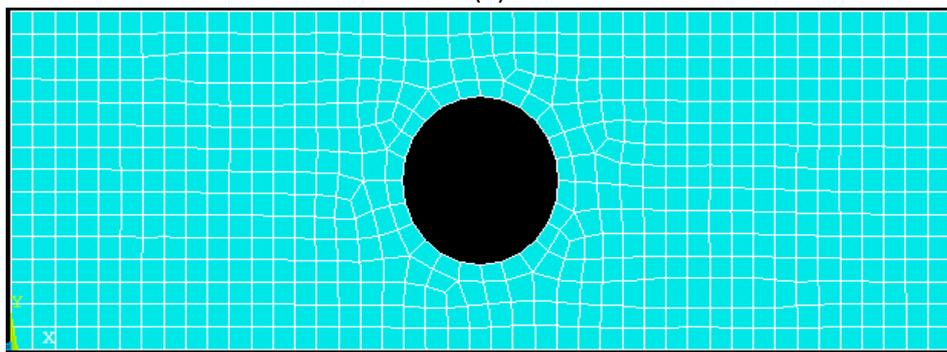
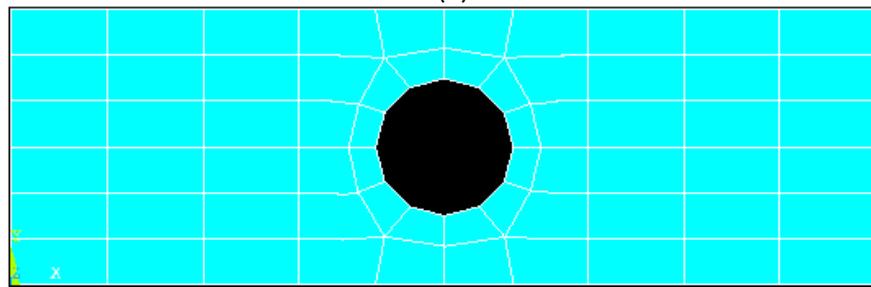
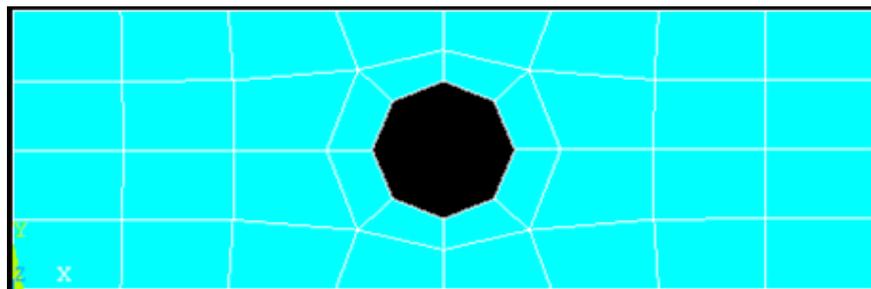
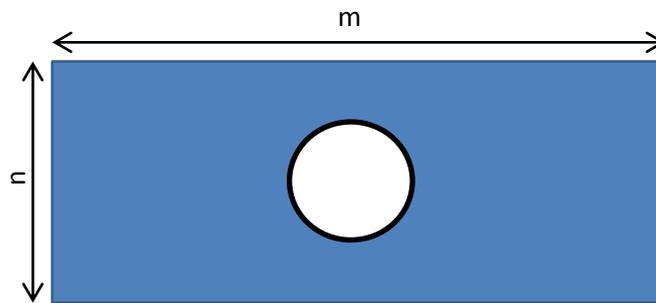


Figure V.18 : Schéma des maillages, (a) maillage 8×4 , (b) maillage 12×4 , (c) maillage 44×15 .

Chapitre V : Applications

Les résultats obtenus (déplacements et contraintes maximales) par les deux programmes (MEFmembrane et ANSYS) pour trois maillages (8x4, 12x4 et 44x15) sont donnés dans le (tableau V.8).



V.19 : L'orientation des maillages utilisés.

Maillage m x n	Résultats			
	MEFmembrane (MATLAB)		ANSYS	
	Déplacements (mm)	Contraintes (MPa)	Déplacements (mm)	Contraintes (MPa)
8x4	0.34	236.74	0.34	279.58
12x4	0.35	293.54	0.35	329.03
44x15	0.35	377	0.36	405.5

Tableau V.8 : Résultat du déplacement (mm) et contrainte (MPa) maximal de la plaque trouée pour $\nu = 0.3$.

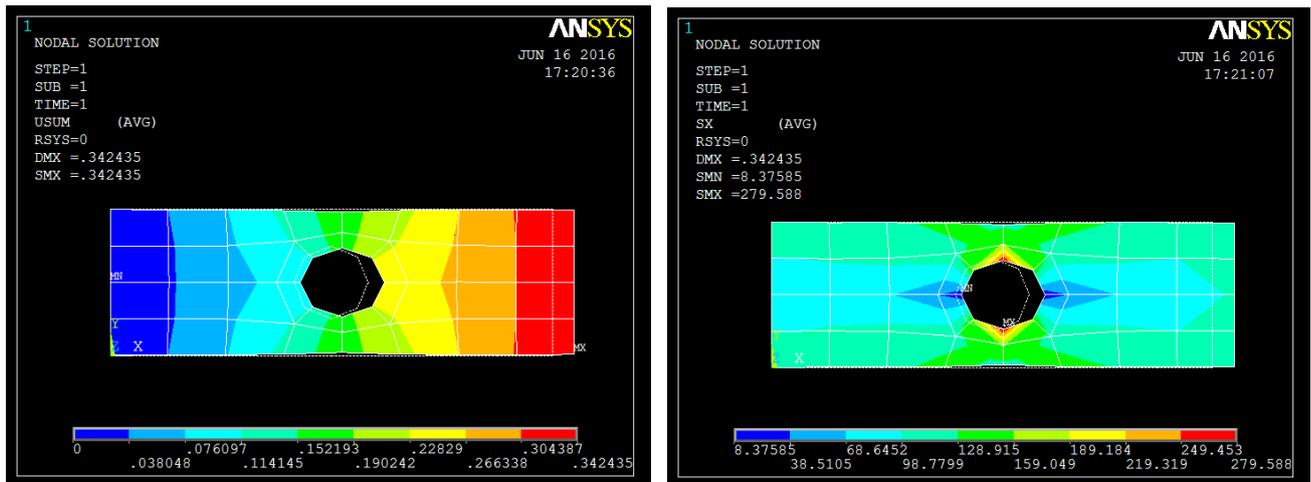


Figure V.20: Déplacements (mm) et contraintes (MPa) de la plaque trouée pour un maillage (8x4), ($\nu = 0.3$) avec ANSYS.

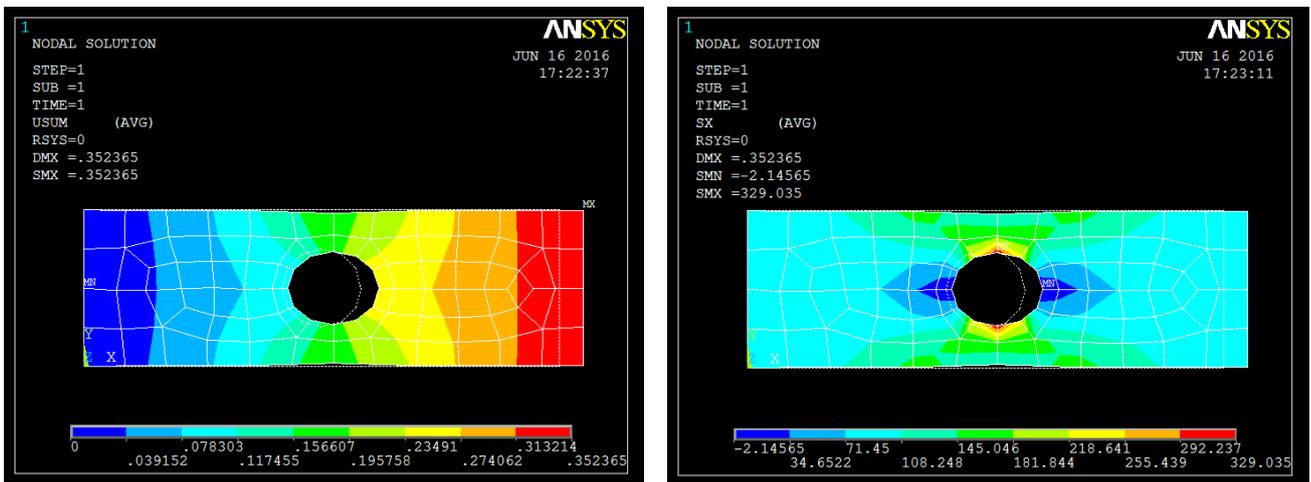


Figure V.21 : Déplacements (mm) et contraintes (MPa) de la plaque trouée pour un maillage (12x4), ($\nu = 0.3$) avec ANSYS.

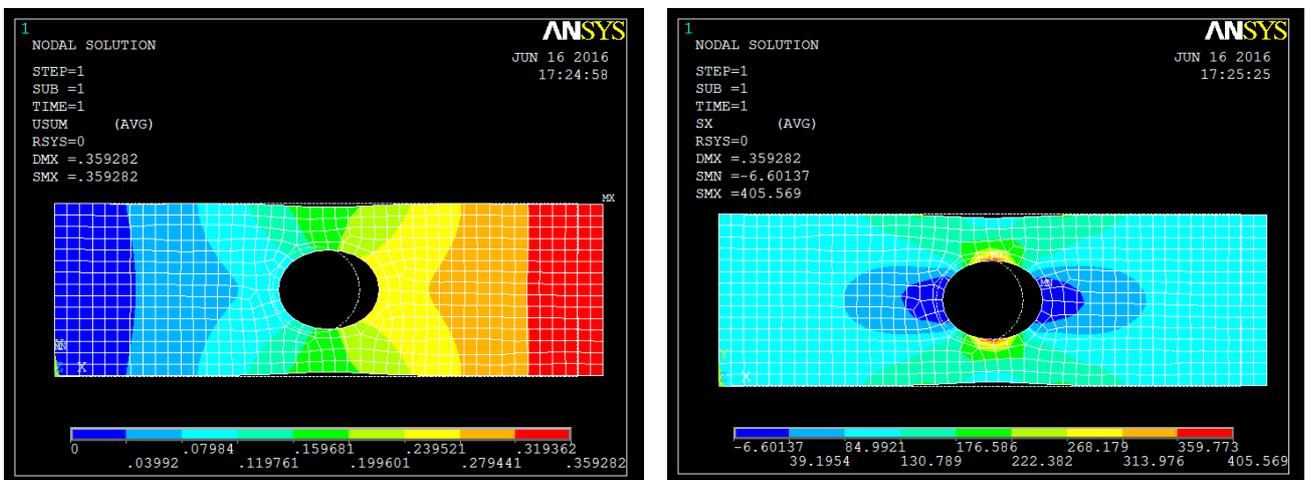


Figure V.22 : Déplacements (mm) et contraintes (MPa) de la plaque trouée pour un maillage (44x15), ($\nu = 0.3$) avec ANSYS.

Chapitre V : Applications

V.3.2.1 Interprétation des résultats

Le tableau V.8 montre que lorsqu'il s'agit des déplacements les résultats sont très proches la valeur retenue est 0.35 mm.

Les résultats en contrainte présentent une différence acceptable qui ne dépasse pas les 7% pour le maillage le plus fin (44*15).

Une concentration des contraintes est observée au bord du trou. Afin de valider notre modèle, nous cherchons à comparer nos résultats par calcul analytique. La contrainte maximale est obtenue en multipliant la contrainte nominale par le facteur d'intensité de contrainte calculé par la formule suivante

$$K_t = 1 - \frac{A - B \frac{2r}{L}}{\frac{h}{r} \frac{r}{L} + C} + 2 \left(\frac{2r}{L} D + 1 \right) e^{-\frac{2r}{L} D} = 2.16$$

A=0.955, B=1.074, C=1.246, D=2.8.

la contrainte nominale est :

$$\sigma_{xx} = q * \frac{h}{t * (h - r)} = 200 \text{ MPa}$$

la contrainte maximale : $\sigma_{max} = K_t * \sigma_{xx} = 2.16 * 200 = 432 \text{ MPa}$.

L'écart entre le programme MEFmembrane et ANSYS est de 7% pour le dernier maillage, MATLAB ne nous donne pas non plus la possibilité d'utiliser des techniques de visualisation.

Enfin, nous constatons que plus nous raffinons le maillage, plus les résultats deviennent similaires. Nous pouvons penser qu'un maillage plus fin donnerait des résultats encore plus précis.

Conclusion générale

Conclusion générale

L'étude du comportement des éléments barres, poutres et membranes sont les bases du domaine de calcul des structures, avec le développement des outils informatique et l'apparition des codes de calcul très performant basé généralement sur la MEF. Le domaine de la modélisation des structures tend à se démocratiser et devient accessible aux non-spécialistes. Le fait de programmer les éléments dans un logiciel type MATLAB permet de comprendre la MEF.

Ce travail, après une présentation de la MEF et les rappels sur la mécanique des solides (chapitre I et II) a permis d'analyser statiquement des éléments des structures (barre, poutre et membrane) par la MEF.

Ce travail, très utile pour notre formation nous a permis de comprendre un certain nombre de choses, entre autre :

a) La MEF est une méthode très universelle pour résoudre les problèmes de l'ingénieur dans différentes domaines, sa bonne compréhension nécessite une sérieuse base mathématique.

b) La MEF repose sur la formulation faible des équations aux dérivées partielles via les fonctions poids et l'intégration par parties, qui nous fournit une réduction d'ordre des dérivées et permet de prendre en compte les conditions aux limites.

c) Le choix des fonctions de formes ainsi que leurs intégration exacte pour formuler les éléments barres et poutres ont démontrés des résultats très fiables.

d) La transformation des charges réparties en charges nodales est nécessaire dans la MEF.

Ainsi, durant notre projet, 3 programmes MATLAB ont été développés :

- Un programme avec des éléments barres qui a donné, pour deux exemples, des résultats satisfaisants et très proches du modèle ANSYS correspondant.
- Un programme avec des éléments poutres qui a également donné, pour deux exemples, des résultats satisfaisants comparables au modèle ANSYS correspondant.
- Un programme avec des éléments membrane pour lequel nous avons réalisé deux modèles de plaque en traction, une plaque pleine et une autre trouée.

Ce second modèle, plus complexe a également montré des résultats très satisfaisants, avec 7% de différence en contrainte maximale par rapport au modèle ANSYS, pour le maillage le plus fin.

Nous pensons que nous pouvons arriver à un résultat encore plus précis avec un maillage raffiné.

Conclusion générale

En perspective, nous pouvons imaginer un développement similaire d'éléments coque (prenant en compte la flexion), ou même le passage à une étape supérieure : la modélisation des phénomènes non-linéaires comme le contact, la plasticité, ou la modélisation de phénomènes transitoires.

Bibliographie

- [1] : Mickael ABBAS, « La méthode des éléments finis isoparamétriques », 2013.
- [2] : A.Quarteroni, F.Saleri et P.Gervasio « Calcul scientifique: cours, exercices corrigés et illustrations en Matlab et Octave » : édition Springer,2011.
- [3] : E.Davalle « Mécanique des structures 1 » LSMS EPFL,2014.
- [4] : Abdelghni Seghir « Méthode des éléments finis », UAMB, 2005-2014.
- [5] : G.Dhatt, G.Touzot et Emmanuel Lefrançois « Méthode des éléments finis, une présentation » : édition hermes, 2004.
- [6] : Yves Debard « Méthode des éléments finis : élasticité plane »,2016-2011.
- [7] : Michel Cazenave, « Méthode des éléments finis approche pratique en mécanique des structures » : édition Dunod, 2010.
- [8] : O.Rahmani et S.Kabdani « Introduction à la méthode des éléments finis pour les ingénieurs » : office des publications universitaire, 1994.
- [9] : François Frey « Analyse des structures et milieux continus. Méthode des éléments finis : édition PPUR, 2001.
- [10] : Lakhdar Bessais « Analyse des structure par M.E.F basée sur l’approche en déformation », UMKB , 2012.
- [11] : Kaushalkumar Kansara « Development of membrane, Plat and Flat Shell Elements in Java », VPI&SU,2004.

Annexes

Toutes les sources des programmes développés sont dans le DVD et dans le site <https://sourceforge.net/projects/sfeatool/>