

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



Université de Biskra

Faculté des Sciences et de la Technologie

Département de Génie Mécanique

Filière : Génie Mécanique

Option: Conception Mécanique Et Productique

Réf:.....

Mémoire de Fin d'Etude

En vue de l'obtention du diplôme de

MASTER

Thème

EVALUATION D'ÉLÉMENTS FINIS VOLUMIQUES DANS
ABAQUS SUR DES CAS TESTS NON LINÉAIRES
GÉOMÉTRIQUES

Présenté par:

Mr. Bouleftour Abderrahmane

Proposé et dirigé par:

Dr. Meftah Kamel

Promotion : Juin 2014

Remerciement

*Mes remerciements à dieu tout puissant pour
Pla volonté, la santé et la patience qu'il m'a donnée toujours.*

*Je tiens à remercier chaleureusement mon encadreur : **Docteur Meftah Kamel** d'avoir accepté de diriger ce travail et de m'avoir prodigué des Conseils scientifiques, ont été précieux et des orientations pour la réalisation de ce travail.*

Des remerciements aux membres du jury, pour l'intérêt et d'avoir accepté de juger ce travail.

A l'ensemble des enseignants qui ont contribué à ma formation ainsi que tout le personnel administratif qui, a participé au bon déroulement de ce travail, les enseignants du département de génie mécanique. Pour sa disponibilité, sa patience et ses conseils qu'ils ont prodigué.

Mes remerciements vont aussi à tous mes parents, mes enseignants, mes amis et mes collègues de la promotion 2014.

A tous ceux qui, par leur encouragement ou leur amitié.

Dédicace

Je dédie ce travail :

*Ma chère mère qui n'a vécu que pour me voir réussir dans ma vie et avoir un
bon statut social*

A mon père

A mes frères et sœurs et à tous ma famille

A tous mes amis et collègues de la promotion 2014

A tous ceux qui m'ont aidé à finir ce mémoire.

Sommaire

Remerciement	i
Dédicace	ii
Sommaire	
Liste des figures	iii
Liste des symboles.....	v
Introduction générale.....	1

Chapitre 1 : Méthode des éléments finis

1.1. Généralités	2
1.2. Démarche éléments finis	3
1.2.1. Discrétisation géométrique	3
1.2.2. Approximation nodale.....	4
1.3. Définition de l'approximation nodale	4
1.3.1. Construction d'une approximation nodale linéaire	4
1.4. Bases polynomiales complètes	4
1.5. Bases polynomiales incomplètes	5
1.6. Approximation nodale de quelques éléments de référence	5
1.6.1.Éléments à une dimension.....	5
1.6.2.Élément triangulaire.....	7
1.6.3.Élément à deux dimensions rectangulaires	8
1.7. Quantités élémentaires	9
1.8. Matrices masse et raideur	9
1.9. Efforts imposés	10
1.10. Assemblage et conditions aux limites	10
1.10.1. Matrices	10
1.10.2. Efforts imposés.....	11
1.10.3. Efforts inconnus	11
1.11. Transformation géométrique	11
1.11.1. Exemples d'éléments de référence classiques	12
1.12. Présentation du type d'élément fini	14

1.12.1 Approximation des déplacements	15
1.12.2. Approximation de la géométrie	16
1.12.3. Déduction de la matrice	16
1.12.4. Détermination de la forme finale de la matrice de rigidité de l'élément	17

Chapitre 2 : Généralités sur le logiciel Abaqus

2.1. Introduction	19
2.2. Généralité sur le code de calcul ABAQUS	20
2.2.1. ABAQUS/Standard	20
2.2.2. ABAQUS/Explicit	20
2.2.3. ABAQUS/CAE	20
2.3. Les différents fichiers pour la simulation sous Abaqus	21
2.3.1. Fichier d'entrée (Input file)	21
2.3.2. Fichier de résultats (output file)	22
2.4. Autres fichiers créés par Abaqus lors de la simulation	22
2.5. Organisation de l'interface Abaqus CAE (complète Abaqus environnement)	22
2.6. Les modules	23
2.6.1. Module part.....	24
2.6.2. Module Property	25
2.6.3. Module Assembly	26
2.6.4. Module Step	27
2.6.5. Module Interaction	28
2.6.6. Module Load	28
2.6.7. Module Mesh	30
2.6.8. Module JoB	30
2.6.9. Module visualization	31
2.6.10. Module sketch	31
2.7. L'arbre du modèle.....	31
2.8. Les éléments dans ABAQUS	32
2.8.1. Degré de liberté (DDL)	32
2.9. Les Types de non-linéarités.....	37

2.9.1. Non-linéarités matérielles.....	37
2.9.2. Non-linéarités géométriques	37
2.9.3. Non-linéarités de contact.....	37

Chapitre 3 : Formulation non linéaire géométrique

3.1. Introduction.....	38
3.2. La description Lagrangienne.....	39
3.2.1. La formulation Lagrangienne totale.....	40
3.2.2. La formulation Lagrangienne actualisée.....	40
3.3. Formulation non linéaire géométrique.....	40
3.3.1. Principe des travaux virtuels.....	40
3.3.2. Formulation Lagrangienne totale.....	41
3.3.3. Approximation des champs de déplacements et de déformations.....	43
3.4. Méthodes numériques de résolution.....	47

Chapitre 4 : Cas tests non linéaires

4.1. Introduction.....	50
4.2. Schéma d'incrément de charge.....	51
4.3. Comparaison numérique en analyse non linéaires.....	53
4.3.1. Poutre console soumise à un effort tranchant.....	53
4.3.2. Coque semi cylindrique	56
4.3.3. Coque cylindrique sans diaphragmes soumise à une force ponctuelle.....	58
Conclusion	61
Références bibliographique	

Liste des figures

Figure 1.1. Erreur de discrétisation géométrique	3
Figure 1.2. Fonction de forme à une dimension	6
Figure 1.3. Fonctions d'interpolation linéaires du triangle	7
Figure 1.4. Fonctions d'interpolation quadratiques du triangle. Les autres sont obtenues par rotation	7
Figure 1.5. Fonction d'interpolation $M1$ du quadrangle. Les autres sont obtenues par rotation	8
Figure 1.6. Transformation géométrique linéaire d'un carré	12
Figure 1.7. Transformations géométriques d'éléments à une dimension avec en haut, l'élément réel et en bas, l'élément parent	12
Figure 1.8. Éléments parents triangulaires et quadrangulaires à deux dimensions	13
Figure 1.9. Éléments parents volumiques à transformation linéaire	13
Figure 1.10. Élément fini tridimensionnel rapporté au système de coordonnées globales	14
Figure 1.11. Élément fini tridimensionnel rapporté au système de coordonnées naturelles	14
Figure 2.1. Structure d'une analyse sous Abaqus [6]	21
Figure 2.2. Organisation d'abaqus [7]	23
Figure 2.3. Réalisation de la mise en données d'un problème	23
Figure 2.4. Module Part	24
Figure 2.5. Création de la géométrie [9]	25
Figure 2.6. propriété Matériaux et section	26
Figure 2.7. Instance et assemblage	27
Figure 2.8. Création de step	27
Figure 2.9. Interaction	28
Figure 2.10. Création des chargements	29
Figure 2.11. Création de maillage	30
Figure 2.12. Lancement de calcul	30
Figure 2.13. Affichage des résultats [9]	31
Figure 2.14. Quelque élément dans Abaqus [10]	32
Figure 2.15. Les types d'intégrations et le nombre de nœuds	33
Figure 2.16. Exemple d'élément poutre	34

Figure 2.17. Éléments solides 1D	35
Figure 2.18. Éléments solides 2D	35
Figure 2.19. Éléments axisymétriques 2D	36
Figure 2.20. Éléments coques et 3D	36
Figure 3.1. Configurations initiale, intermédiaire et finale d'un corps solide	39
Figure 4.1. Schéma d'incrémentation de charge automatique [21]	52
Figure 4.2. Poutre console en non linéaire géométrique	54
Figure 4.3. Résultats de la poutre consol en non linéaire géométrique avec les éléments C3D8 et C3D20	54
Figure 4.4. Configuration initial ($P = 0$) de la poutre	55
Figure 4.5. Déformation de la poutre consol ($P = P_{\max}$)	55
Figure 4.6. Coque semi cylindrique	56
Figure 4.7. Courbes force-déplacement de la coque pour les deux éléments volumiques C3D20 et C3D8	56
Figure 4.8. Déformation de la structure [18]	57
Figure 4.9. Déformation de la structure avec élément volumique C3D20	57
Figure 4.10. Coque cylindrique soumise à une force ponctuelle. Géométrie, chargement et Conditions aux limites [18]	59
Figure 4.11. Déflexion WA de la coque cylindrique soumise à une charge ponctuelle. Résultats de l'élément C3D20	59
Figure 4.13. Maillage déformé du cylindre ouvert sous la charge maximale	60
Figure 4.14. Maillage déformé du cylindre ouvert sous la charge maximale maillage 24×36 [18]	60

Liste des symboles et abréviation

Notation matricielle

$[x]$	Matrice
$\{x\}$	Vecteur colonne
$\langle x \rangle$	Vecteur ligne
$[x]^T$	Transposée d'une matrice
$[x]^{-1}$	Inverse d'une matrice
$\det([x])$	Déterminant d'une matrice

Symboles spéciaux

D_e	Le domaine élémentaire
(M)	la valeur de la fonction approchée en tout point
N_i	les fonctions de forme
u_n	variables nodales relatives aux nœuds d'interpolation
Φ	Une base de fonctions indépendantes
A	le vecteur des paramètres de l'approximation
U	la valeur du champ des variables
L	Longueur de l'élément
$(1,s)$	la base de forme à une dimension
$(1,s,t)$	la base d'interpolation linéaire du triangle
$(1, s, t, st)$	la base polynomiale

\vec{u}	Déplacement
$\delta\vec{u}$	Le champ des déplacements virtuels
$[\varepsilon]$	Tenseur de déformation
$[\sigma]$	Tenseur de contrainte
$[B]$	Matrice relations entre les déplacements et les déformations relatifs
$[K_e]$	Matrice raideur élémentaire
δT_{i_e}	travail virtuel élémentaire
M_e	matrices élémentaires masse
F_i	Efforts inconnus
F_d	Efforts imposés
\mathbf{Ng}	Transformation géométrique
U_{xi}	le déplacement du nœud
S	la surface externe
\vec{U}	Vecteur des déplacements
E	Module de youg
ν	Coefficient de Poisson
λ, G	Coefficients de Lamé
$\{R\}$	Vecteur de forces résiduelles
$\frac{\partial}{\partial x}$ ou « _{,x} »	Dérivation partielle par rapport à x
Dx	Différentielle ou incrément
δx	Variation

Δx	Accroissement
δ_{ij}	Symbole de Kronecker
i, j, k, l	Indices latins prennent des valeurs 1, 2, 3
$[K]$	Matrice de rigidité
$[M]$	Matrice masse
ξ, η, ζ	Coordonnées naturelles d'un élément fini
u_i, v_i, w_i	Degrés de liberté de déplacements
C_{ijkl}	Composantes de la matrice de comportement en élasticité linéaire
P_{MAX}	La charge maximale
NINC	conteurs d'itération conteur d'incrément de charge
NINTER	conteurs d'itération
MEF	Méthodes des éléments finis
D.L.A	Description Lagrangienne actualisée
PTV	principe des travaux virtuels

Introduction général

Introduction générale

Depuis les premiers jours, la recherche sur le développement des nouveaux éléments finis a servi cinq principaux objectifs : (1) améliorer les performances dans les problèmes dominés par la flexion ; (2) éviter les différents phénomènes de verrouillages (membrane, cisaillement,...) ; (3) éviter le blocage volumétrique dans les matériaux incompressibles ou presque incompressibles ; (4) réduire la sensibilité de l'élément à la distorsion du maillage ; et (5) développer des éléments plus efficaces avec une précision suffisante (par exemple les éléments stabilisés). Pour répondre à ces exigences, le développement des éléments volumiques à précision améliorée a été traditionnellement l'un des principaux objectifs de recherches d'éléments finis.

Pour examiner ou démontrer la performance des nouveaux modèles éléments finis volumiques ou l'efficacité des nouvelles méthodes de résolutions des problèmes non linéaires, des cas tests sont souvent utilisés et les résultats sont comparés à des solutions de référence. La plupart des solutions de référence sont des solutions numériques dans le cas des solides 3D. Dans ce contexte, des cas tests sont évalués pour l'analyse non linéaire géométrique des solides 3D utilisant le code éléments finis ABAQUS.

Cette étude a abouti à la rédaction de ce manuscrit que nous avons décidé de scinder en quatre chapitres, précédés par une **introduction générale** et clôturés par une **conclusion générale**. Après avoir établi, dans le **premier chapitre**, une description générale du code de calcul Abaqus. Ensuite, nous présenterons dans le **deuxième chapitre** le principe général de la méthode des éléments finis, nous présenterons une synthèse de la formulation des éléments finis volumiques. Le **troisième chapitre** sera l'occasion de présenter la formulation des éléments volumiques classiques en non linéarité géométrique de grands déplacements et petites déformations. Deux éléments non linéaires (hexaédriques à 8 nœuds et à 20 nœuds) sont formulés en intégrant les effets non linéaires géométriques. Nous établissons les équations gouvernantes du problème de grands déplacements en utilisant la formulation lagrangienne totale associée à la méthode de résolution de Newton-Raphson. Le **quatrième chapitre** sera l'occasion de présenter les résultats des simulations numériques obtenus avec les modèles éléments finis volumiques C3D8 et C3D20 présentés dans Abaqus. Ces cas tests en grands déplacements permettant de valider les éléments finis volumiques C3D8 et C3D20 en non linéaire géométrique.

CHAPITRE 1

Généralité sur logiciel Abaqus

1.1. Généralités

Les codes éléments finis font maintenant partie des outils couramment utilisés lors de la conception et à l'analyse des produits industriels. Les outils d'aide à la modélisation devenant de plus en plus perfectionnés, l'utilisation de la méthode des éléments finis s'est largement développée et peut sembler de moins en moins une affaire de spécialistes. Si l'utilisation de la méthode se démocratise de par la simplicité croissante de mise en œuvre, la fiabilité des algorithmes et la robustesse de la méthode, il reste néanmoins des questions essentielles auxquelles l'ingénieur devra répondre s'il veut effectuer une analyse par éléments finis dans de bonnes conditions :

- formaliser les non dits et les réflexions qui justifient les choix explicites ou implicites de son analyse du problème
- évaluer la confiance qu'il accorde aux résultats produits
- analyser les conséquences de ces résultats par rapport aux objectifs visés.

L'objectif de cette partie est de présenter les principes de base de cette méthode en insistant sur l'enchaînement des tâches (démarche et hypothèses associées) qui assurent la cohérence du processus de calcul. Ces connaissances vous seront utiles pour maîtriser les deux principales difficultés de mise au point d'un modèle numérique :

- problèmes préliminaires à la phase de calcul.
- problèmes liés à l'exploitation des résultats et le retour à la conception.

Il ne faut pas perdre de vue que l'analyse des résultats nécessite une bonne compréhension des différentes étapes mathématiques utilisées lors de l'approximation, pour pouvoir estimer l'erreur du modèle numérique par rapport à la solution exacte du problème mathématique.

Sans oublier que le modèle numérique ne peut fournir que des résultats relatifs aux informations contenues dans le modèle mathématique qui découle des hypothèses de modélisation.

Nous nous limiterons à la présentation de modèles élémentaires utilisés dans le cadre des théories linéaires. Bien que simples ces modèles permettent déjà de traiter un grand nombre d'applications liées aux problèmes de l'ingénieur. Du point de vue pédagogique, ils sont suffisamment complexes pour mettre en avant les difficultés de mise en œuvre de la méthode.

L'idée fondamentale de cette méthode est de discrétiser le problème en décomposant le domaine matériel à étudier en éléments de forme géométrique simple. Sur chacun de ces éléments il sera plus simple de définir une approximation nous permettant d'appliquer les méthodes présentées dans la première partie de ce cours. Il ne reste alors qu'à assembler les

formes matricielles élémentaires pour obtenir les équations relatives à la structure à étudier. C'est sous cette forme pragmatique qu'elle est utilisée par les ingénieurs, et que nous allons maintenant l'aborder [1].

1.2. Démarche éléments finis

Les principales étapes de construction d'un modèle éléments finis, qui sont détaillées par la suite, sont les suivantes :

- Discrétisation du milieu continu en sous domaines
- Construction de l'approximation nodale par sous domaine
- Calcul des matrices élémentaires correspondant à la forme intégrale du problème
- Assemblage des matrices élémentaires
- prise en compte des conditions aux limites
- Résolution du système d'équations

1.2.1. Discrétisation géométrique

Cette opération consiste à procéder à un découpage du domaine continu en sous domaines :

$$\text{Tel que} \quad D = \sum_{e=1}^{ne} D_e \quad \lim_{e \rightarrow 0} (\cup_e D_e) = D \quad (1.1)$$

Il faut donc pouvoir représenter au mieux la géométrie souvent complexe du domaine étudié par des éléments de forme géométrique simple. Il ne doit y avoir ni recouvrement ni trou entre deux éléments ayant une frontière commune.

Lorsque la frontière du domaine est complexe, une erreur de discrétisation géométrique est Inévitable. Cette erreur doit être estimée, et éventuellement réduite en modifiant la forme ou en diminuant la taille des éléments concernés comme proposé sur la figure 1.1 Sur chaque élément nous allons chercher à définir une approximation de la fonction solution.

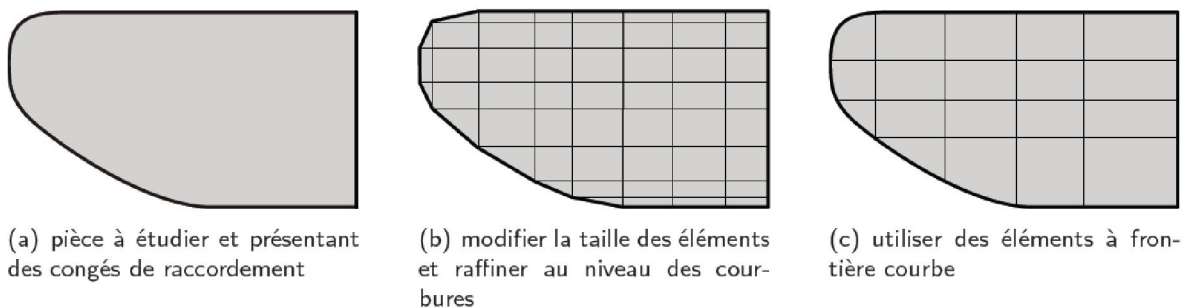


Figure 1.1 : Erreur de discrétisation géométrique

1.2.2. Approximation nodale

La méthode des éléments finis est basée sur la construction systématique d'une Approximation u^* du champ des variables u par sous domaine. Cette approximation est construite sur les valeurs approchées du champ aux nœuds de l'élément considéré, on parle de représentation nodale de l'approximation ou plus simplement d'approximation nodale.

1.3. Définition de l'approximation nodale

L'approximation par éléments finis est une approximation nodale par sous domaines ne faisant intervenir que les variables nodales du domaine élémentaire D_e :

$$\forall M \in D_e, u(M) = N(M)u_n^* \quad (1.2)$$

Où $u^*(M)$ représente la valeur de la fonction approchée en tout point M de l'élément et N , la matrice ligne des fonctions d'interpolation de l'élément u_n variables nodales relatives aux nœuds d'interpolation de l'élément.

1.3.1. Construction d'une approximation nodale linéaire

L'interpolation nodale est construite à partir d'une approximation générale :

$$\forall M, u^*(M) = \Phi(M)a \quad (1.3)$$

Φ Est une base de fonctions connues indépendantes, en général une base polynomiale et a , le vecteur des paramètres de l'approximation aussi dits paramètres généralisés, qui n'ont pas de signification physique.

1.4. Bases polynomiales complètes

- une dimension
 - Linéaire $(1, x)$: deux variables
 - Quadratique $(1, x, x^2)$
- deux dimensions :
 - Linéaire $(1, x, y)$: trois variables
 - Quadratique $(1, x, y, x^2, xy, y^2)$: six variables
- trois dimensions :
 - linéaire $(1, x, y, z)$: quatre variables
 - quadratique $(1, x, y, z, x^2, xy, y^2, xz, z^2, yz)$: dix variables

Pour utiliser une base polynomiale complète, le nombre de termes doit être égal au nombre de variables nodales à identifier. Si l'on ne peut pas utiliser un polynôme complet, le meilleur choix consiste à respecter la symétrie des monômes conservés.

1.5. Bases polynomiales incomplètes

- deux dimensions : « bi - linéaire » $(1, x, y, xy)$: quatre variables
- trois dimensions : « tri - linéaire » $(1, x, y, z, xy, xz, yz, xyz)$: huit variables

En identifiant aux nœuds l'approximation u^* à la valeur du champ de variables u , nous pouvons exprimer les paramètres généralisés a en fonction des variables nodales u_n .

$$u_n = u^*(M_n) = \Phi(M_n) a \quad (1.4)$$

Soit, par inversion du système total :

$$a = T u_n \quad (1.5)$$

Pour éviter des erreurs de modèle trop importantes, la matrice à inverser doit être bien conditionnée.

Ce conditionnement est lié au choix de la base polynomiale et à la géométrie des

Éléments. En reportant ce résultat dans l'approximation nous obtenons la matrice des fonctions d'interpolation :

$$N(M) = \Phi(M)T \quad (1.6)$$

1.6. Approximation nodale de quelques éléments de référence

Un élément de référence est un élément de forme géométrique simple, à frontière rectiligne par exemple, pour lequel l'approximation nodale est construite en suivant la démarche analytique précédente. Le passage de l'élément de référence à l'élément réel sera réalisé par une transformation géométrique. Nous entendons par élément réel un élément quelconque du domaine discrétisé.

Deux grandes familles d'éléments sont souvent présentées :

- les éléments de type Lagrange
- les éléments de type Hermite

Pour les éléments de type Lagrange, on augmente le nombre de nœuds en conservant une seule variable nodale. Pour les éléments de type Hermite on augmente le nombre de variables nodales, en retenant par exemple les valeurs des dérivées du champ aux nœuds. L'élément poutre présenté dans le chapitre suivant fait parti de la famille de l'Hermite [1].

1.6.1.Éléments à une dimension

La base de fonction linéaire illustrée sur la figure 1.2(a) s'écrit avec $s \in [0,1]$

$$\begin{aligned} N_1(s) &= L_1 = 1 - s \\ N_2(s) &= L_2 = s \end{aligned} \quad (1.7)$$

Cette base est utilisée pour les éléments barres et génère une discontinuité au niveau des champs de déformations et de contraintes au passage d'un élément à son voisin. Une base un peu plus riche, constituée de polynômes d'ordre 2 peut aussi être utilisée :

$$\begin{aligned} N_1(s) &= L_1 A = (2L_1 - 1) \\ N_2(s) &= 4L_1 L_2 \\ N_3(s) &= L_2(2L_2 - 1) \end{aligned} \quad (1.8)$$

Ces fonctions de forme sont schématisées sur la figure 1.2(b). Le passage à l'ordre supérieur donne la base de la figure 1.2(c) où seules N_1 et N_2 sont illustrées : les deux autres fonctions N_3 et N_4 sont respectivement les symétriques de N_1 et N_2 par rapport à $S=1/2$

$$\begin{aligned} N_1(s) &= \frac{L_1}{2}(3L_1 - 1)(3L_1 - 2) \\ N_2(s) &= \frac{9}{2}L_1 L_2(3L_1 - 1) \\ N_3(s) &= \frac{9}{2}L_1 L_2(3L_2 - 1) \\ N_4(s) &= \frac{L_2}{2}(3L_2 - 1) \end{aligned} \quad (1.9)$$

L'élément associé est construit avec quatre nœuds et une variable par nœud. Il est possible, avec la même base polynomiale, de construire un élément à deux nœuds ayant deux variables par nœud, c'est un élément de type l'Hermite illustré sur la figure 1.2(d) pour N_1 et N_2 : de la même manière que précédemment, les fonctions N_3 et N_4 se trouvent par symétrie.

Si nous utilisons comme variables nodales le champ et sa dérivée première, nous obtenons les fonctions d'interpolation de l'élément poutre présenté dans la sous-section 3.2.1 :

$$\begin{aligned} N_1(s) &= 1 - 3s^2 + 2s^3 \\ N_2(s) &= s - 2s^2 + s^3 \\ N_3(s) &= 3s^2 - 2s^3 \\ N_4(s) &= -s^2 + s^3 \end{aligned} \quad (1.10)$$

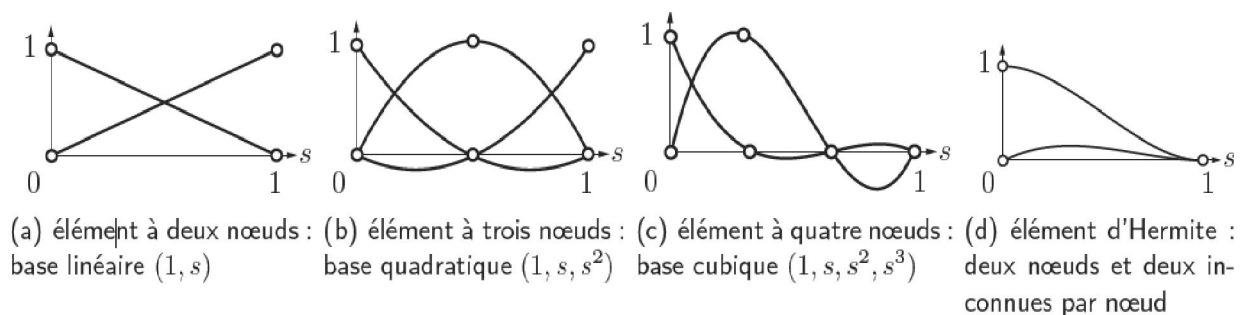


Figure 1.2. Fonction de forme à une dimension

1.6.2.Élément triangulaire

Pour ce type d'élément, l'approximation utilise la base polynomiale linéaire $(1, s, t)$. L'élément de référence, aussi dit parent, est un triangle rectangle à trois nœuds de type

« T3 ».figure1.3 L'approximation quadratique quant à elle utilise la base $(1, s, t, s^2, st, t^2)$.

L'élément de référence est un triangle rectangle à six nœuds de type « T6 ».

Posons

$$L_1 = 1 - s - t$$

$$L_2 = s \tag{1.11}$$

$$L_3 = t$$

Pour :

– les trois nœuds sommet $i = 1, 2, 3$, les fonctions de forme s'écrivent :

$$N_i = L_i(2L_i - 1) \tag{1.12}$$

les trois nœuds d'interface $i = 1,2,3$

$$N_{i+3} = 4L_jL_k \text{ Pour } j \neq i \quad k \neq i,j \tag{1.13}$$

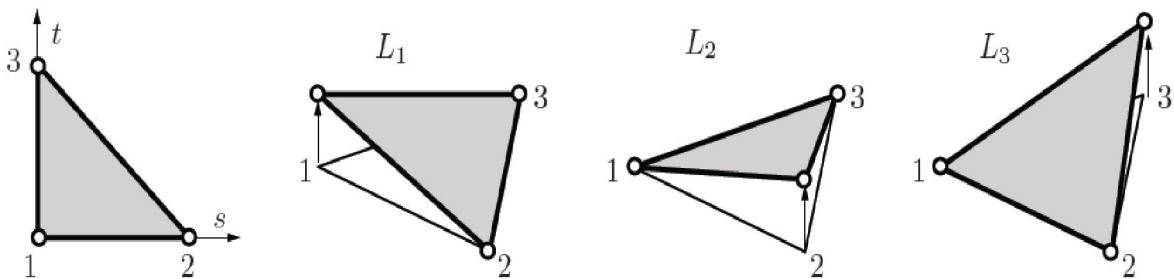


Figure 1.3 : Fonctions d'interpolation linéaires du triangle

La figure 1.4 donne une représentation de deux des fonctions d'interpolation. Les autres s'obtiennent par permutation des indices.

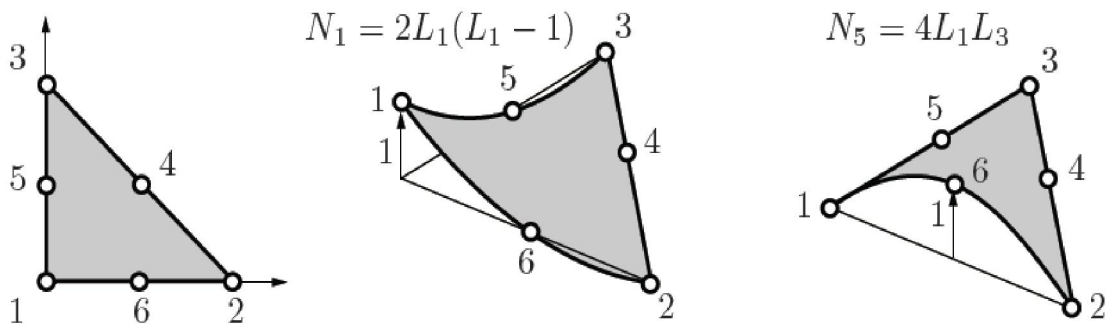


Figure 1.4 : Fonctions d'interpolation quadratiques du triangle. Les autres sont obtenues par rotation

1.6.3.Élément à deux dimensions rectangulaires

L'approximation bi-linéaire est déduite de la base polynomiale $(1, s, t, st)$ sur $(s, t) \in [-1, 1]$. L'élément de référence est un carré à quatre nœuds de type « Q4 ». Les fonctions d'interpolation sont :

$$\begin{aligned} N_1 &= \frac{1}{4}(1 - S)(1 - t) \\ N_2 &= \frac{1}{4}(1 + S)(1 - t) \\ N_3 &= \frac{1}{4}(1 + S)(1 + t) \\ N_4 &= \frac{1}{4}(1 - S)(1 + t) \end{aligned} \quad (1.14)$$

Sur la figure 1.5, seule la fonction N_1 est représentée, les autres s'obtenant par permutation. De la même façon, on peut construire, à partir d'une base polynomiale complète, les fonctions d'interpolation des éléments rectangulaires à neuf nœuds, pour une approximation quadratique et à seize nœuds pour une approximation polynomiale cubique. Ces éléments ont respectivement un et quatre nœuds internes.

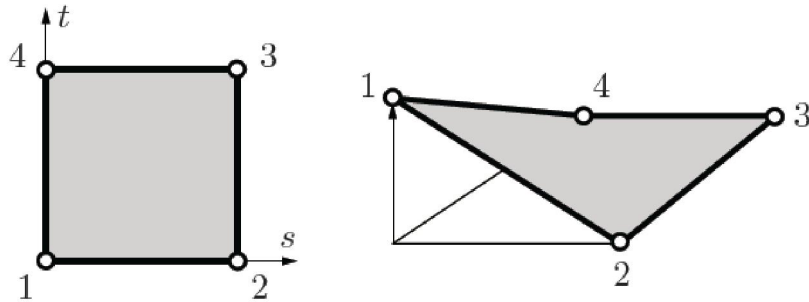


Figure 1.5 : Fonction d'interpolation N_1 du quadrangle. Les autres sont obtenues par rotation

Du point de vue pratique, on construit des éléments ayant un minimum de nœuds internes, car ces nœuds ne sont pas connectés aux nœuds des autres éléments. On utilise donc des bases polynomiales incomplètes mais symétriques. Le Q8 est construit à partir de la base $(1, s, t, s^2, st, t^2, s^2t, st^2)$ et le Q12 est construit à partir de la base $(1, s, t, s^2st, t^2, s^3, s^2t, t^2s, t^3, s^3t, st^3)$ [2].

1.7. Quantités élémentaires

Afin de présenter la démarche générale utilisée pour construire les formes matricielles et vectorielles sur chaque élément, nous utiliserons comme point de départ la forme intégrale du Principe des Travaux Virtuels associée à un problème de mécanique des structures. Cette forme intégrale est de même type que celles pouvant être déduites des méthodes variationnelles et la généralisation à des problèmes de physique est donc simple.

1.8. Matrices masse et raideur

Soit la forme intégrale du PTV

$$\forall \delta \vec{u}, \int_D P \vec{u} \delta \vec{u} dV = - \int_D \bar{\delta} : \delta \epsilon dV + \int_D \vec{f} \cdot \delta \vec{u} dV + \int_{\delta DT} \delta u dS \quad (1.15)$$

Sur chaque élément nous utilisons l'approximation nodale pour exprimer le champ des déplacements \vec{u} et le champ des déplacements virtuels $\delta \vec{u}$. Ainsi le produit scalaire s'écrit maintenant

$$\vec{u}(M) \cdot \delta \vec{u}(M) = \delta u_n^T N(M)^T N(M) u_n \quad (1.16)$$

d'où le premier terme :

$$\int_{D_e} P \vec{u} \delta \vec{u} dV = \delta u_n^T M_e u_n \quad (1.17)$$

$$\text{Avec } M_e = \int_{D_e} N(M)^T \rho N(M) dV \quad (1.18)$$

Matrice masse élémentaire. Pour exprimer le second terme les deux tenseurs sont représentés par des vecteurs nous permettant de remplacer le produit doublement contracté par un simple produit scalaire.

$$\bar{\epsilon} \rightarrow \epsilon = [\epsilon_{xx} \quad \epsilon_{yy} \quad \epsilon_{yy} \quad 2\epsilon_{xy} \quad 2\epsilon_{xz} \quad 2\epsilon_{yz}]^T \quad (1.19)$$

$$\bar{\sigma} \rightarrow \sigma = [\sigma_{xx} \quad \sigma_{yy} \quad \sigma_{zz} \quad \sigma_{xy} \quad \sigma_{xz} \quad \sigma_{yz}]^T$$

De plus le vecteur des déformations s'exprime en fonction du champ des déplacements. Ces relations géométriques font apparaître des opérateurs différentiels appliqués à \vec{u} , que nous notons sous forme matricielle :

$$\epsilon(M) = LN(M)u_n = B(M)u_n \quad (1.20)$$

D'où le second terme, écrit dans la base de discrétisation :

$$\int_{D_e} \bar{\sigma} : \bar{\delta \epsilon} dV = \delta u_n^T K_e u_n \quad (1.21)$$

Avec

$$K_e = \int_{D_e} B(M)^T D(M) B(M) dV \quad (1.22)$$

Matrice raideur élémentaire. Il nous reste à exprimer le travail virtuel des efforts. En pratique, on considère d'une part les efforts donnés et d'autre part les efforts inconnus qui sont les efforts nécessaires pour assurer les liaisons cinématiques.

Sur chaque élément, nous utilisons l'approximation du champ de déplacement pour exprimer le travail virtuel de ces efforts.

1.9. Efforts imposés

Leur travail virtuel élémentaire est de la forme :

$$\delta T d_e = \int_{D_e} \vec{f} d. \delta \vec{u} dV + \int_{\partial D_e} \vec{T}_d . \delta \vec{u} dS \quad (1.23)$$

D'où le travail virtuel discrétisé :

$$\delta T_{i_e} = \delta u_n^T F_{i_e} \quad (1.24)$$

En pratique les efforts inconnus représentent les actions mécaniques extérieures à l'élément considéré. On y trouve les efforts de liaison entre les éléments, et éventuellement pour les éléments de frontière les efforts associés aux liaisons cinématiques de la structure.

Comme nous le verrons lors de l'assemblage des équations, les nœuds internes non chargés sont des systèmes mécaniques en équilibre, ce qui entraîne que le torseur des actions mécaniques de tous les efforts élémentaires des éléments connectés à un même nœud est nul. Reportons dans la forme intégrale les résultats obtenus pour chaque élément, nous obtenons une équation matricielle de la forme :

$$\forall D_e, M_e \ddot{u}_n + K_e u_n = F_{d_e} + F_{i_e} \quad (1.25)$$

1.10. Assemblage et conditions aux limites

Les règles d'assemblage sont définies par la relation :

$$D \approx \sum_{e=1}^{ne} D_e \quad (1.26)$$

1.10.1. Matrices

L'assemblage des matrices élémentaires masse M_e et raideur K_e s'effectue selon les mêmes règles. Ces règles sont définies par sommation des termes correspondant au travail virtuel calculé pour chaque élément.

$$\sum_{e=1}^{ne} \delta u_n^T M_e \ddot{u}_n = \delta U^T M \ddot{U} \text{ et } \sum_{e=1}^{ne} \delta u_n^T k_e u_n = \delta U^T K U \quad (1.27)$$

Cette opération traduit simplement que la forme quadratique associée à l'ensemble du domaine est la somme des formes quadratiques des sous-domaines. Elle consiste à « ranger » 1 dans une matrice globale, les termes des matrices élémentaires. La forme de cette matrice dépend bien évidemment de l'ordre dans lequel sont définies les variables globales de U.

1.10.2. Efforts imposés

L'assemblage ne pose pas de problème, il est défini par sommation des termes correspondant au travail virtuel calculé pour chaque élément :

$$\sum_{e=1}^{ne} \delta u_n^T F_{d_e} = \delta U^T F_d \quad (1.28)$$

1.10.3. Efforts inconnus

L'assemblage peut être mené de façon identique. Cependant, si les liaisons entre les éléments sont parfaites la somme des efforts inconnus aux nœuds internes de la structure est nulle. Nous pouvons en tenir compte pour simplifier l'expression du travail virtuel des efforts inconnus, en ne calculant que le travail virtuel des efforts correspondants aux liaisons cinématiques imposées à la structure, et à celui des liaisons non parfaites. Après assemblage, nous obtenons la forme matricielle du principe des travaux virtuels :

$$M\ddot{U} + KU = F_d + F_i \quad (1.29)$$

Sous cette forme, nous avons N équations pour $N + P$ inconnues. Pour résoudre, il faut tenir compte des P conditions aux limites cinématiques associées aux P composantes inconnues du Vecteur F_i^3 .

1.11. Transformation géométrique

Tout élément réel peut être défini comme l'image par une transformation géométrique d'un élément parent dit de référence pour lequel les fonctions d'interpolation sont connues. À l'image de la figure 2.6, la transformation géométrique définit les coordonnées (x, y, z) de tout point de l'élément réel à partir des coordonnées (s, t, u) du point correspondant de l'élément de référence [3] soit :

Un même élément de référence permettra de générer une classe d'éléments réels. À chaque Élément réel correspond une transformation géométrique différente, cette transformation devant être une bijection. Chaque transformation géométrique dépend des coordonnées des nœuds géométriques de l'élément réel. Pour les éléments les plus simples, la transformation est identique pour chaque coordonnée, et utilise une base de fonctions polynomiales.

Si les nœuds d'interpolation et les nœuds géométriques sont confondus, les fonctions de Transformation géométrique \mathbf{N}_g seront identiques aux fonctions d'interpolation \mathbf{N} . Ces éléments sont dits isoparamétriques. En résumé, la transformation géométrique est définie par

$$x = N_g^T(s, t, u)X_n \quad y = N_g^T(s, t, u)Y_n \quad z = N_g^T(s, t, u)Z_n \quad (1.30)$$

Avec x_n, y_n, z_n coordonnées des nœuds de l'élément réel et $\mathbf{N}_g(s, t, u)$ rassemblant les fonctions de la transformation géométrique.

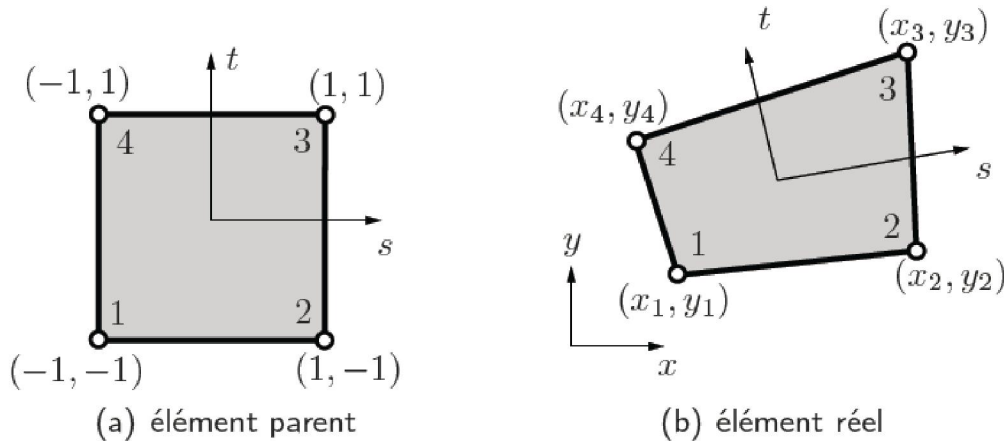


Figure 1.6 : Transformation géométrique linéaire d'un carré

1.11.1. Exemples d'éléments de référence classiques

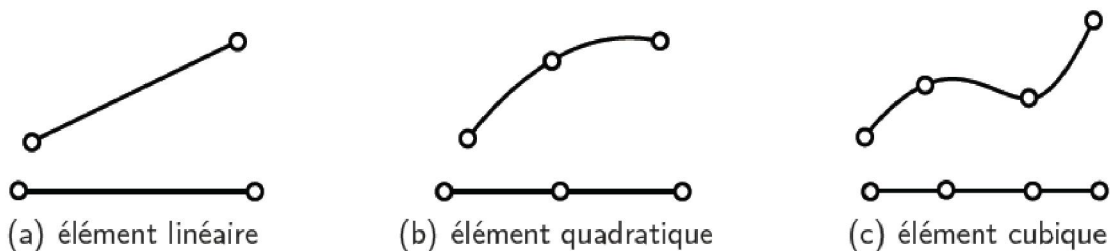


Figure 1.7 : Transformations géométriques d'éléments à une dimension avec en haut, l'élément réel et en bas, l'élément parent

Les transformations géométriques de la figure 1.7 utilisent les fonctions d'interpolation linéaire, quadratique et cubique définies précédemment. Pour des éléments à deux ou trois dimensions les transformations géométriques conduisent respectivement à des frontières linéaires, quadratiques ou cubiques. La figure 1.7 donne la position des nœuds pour les classes d'éléments triangulaires et quadrangulaires.

La figure 1.8 représente les trois classes d'éléments volumiques associés à une transformation linéaire. Pour chaque classe nous définirons les éléments quadratiques et

cubiques en plaçant les nœuds d'interface respectivement au milieu et au tiers des cotés, comme indiqué par la figure 1.9 Les codes éléments finis utilisent ces éléments et bien d'autres plus complexes.

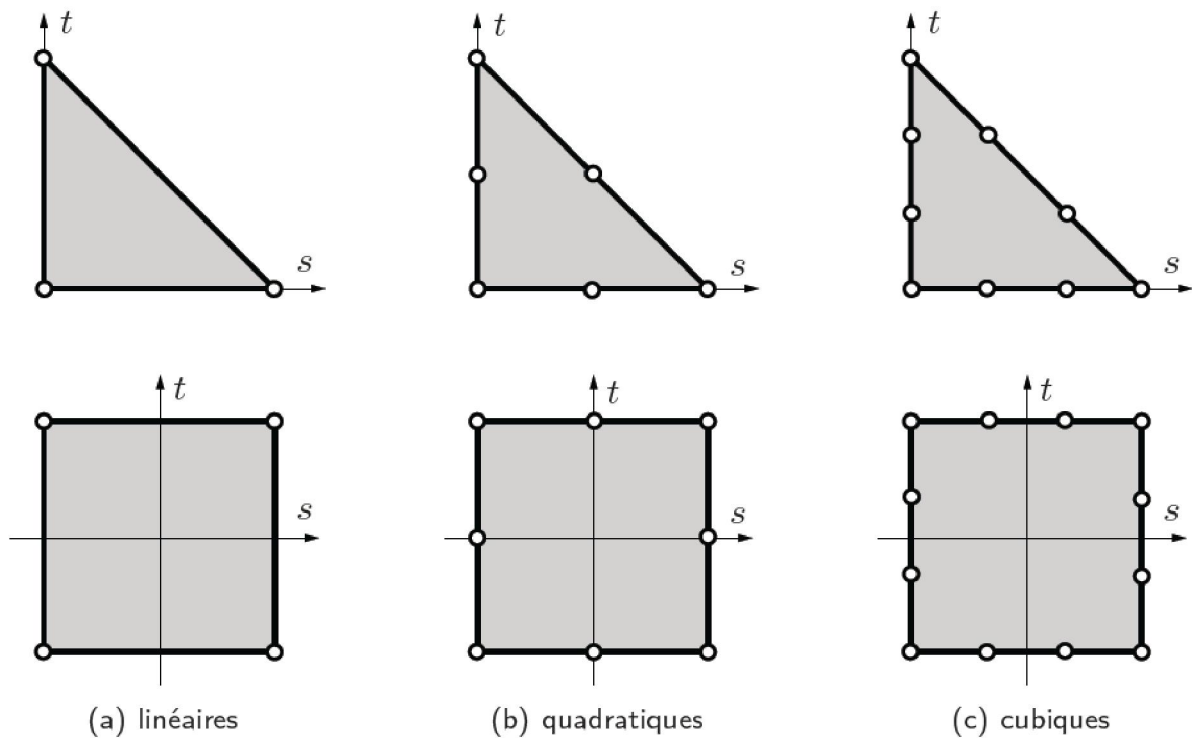


Figure 1.8 : Éléments parents triangulaires et quadrangulaires à deux dimensions

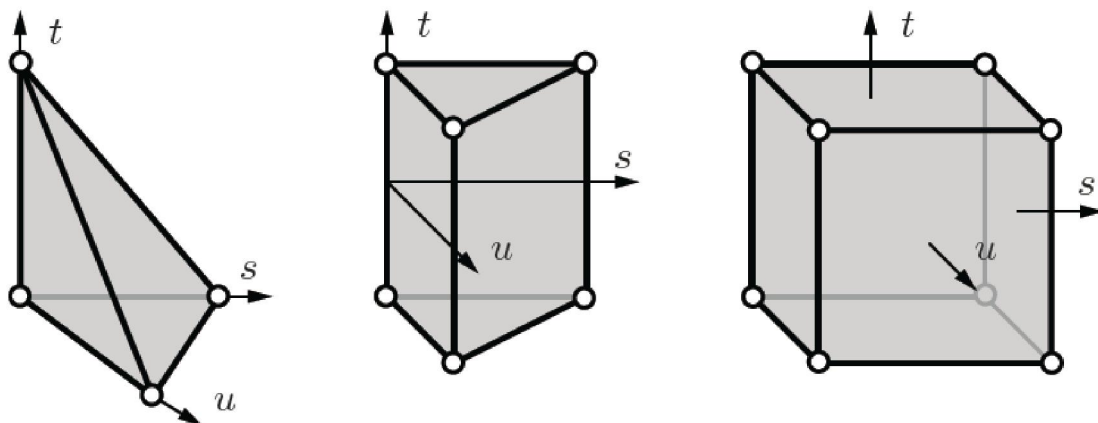


Figure 1.9 : Éléments parents volumiques à transformation linéaire

1.12. Présentation de l'élément fini hexaédrique :

L'élément pris en compte est un élément tridimensionnel figure 1.10 pour lequel est présentée la méthodologie de calcul de la matrice de rigidité, utilisant un élément par couche.

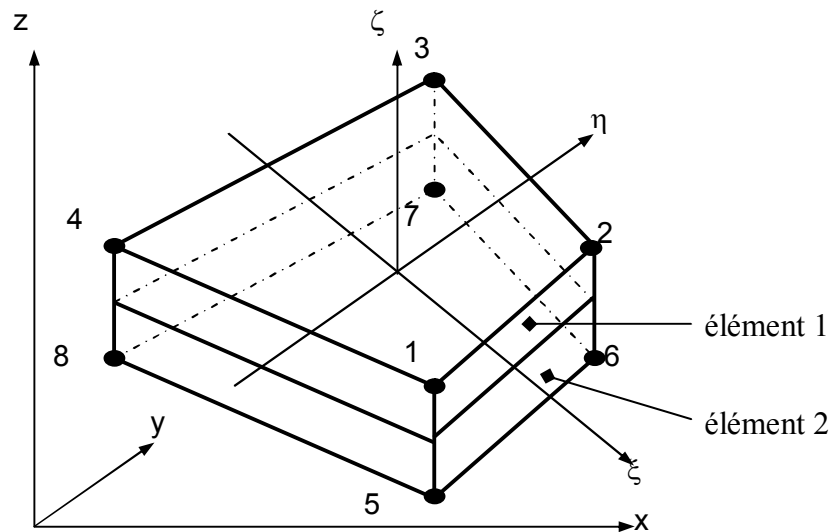


Figure 1.10 : Élément fini tridimensionnel rapporté au système de coordonnées globales

Celui-ci a huit nœuds, trois degrés de liberté pour chaque nœud (trois translations) et rapporté au système de coordonnées naturel $O\xi\eta\zeta$, figure 1.11 Par rapport à son système naturel de coordonnées, l'élément devient un cube ayant les dimensions sur les trois directions comprises entre -1 et 1. C'est un élément isoparamétrique, donc il utilise les mêmes fonctions d'interpolation, aussi bien pour l'approximation du champ de déplacements que pour la description de sa géométrie.

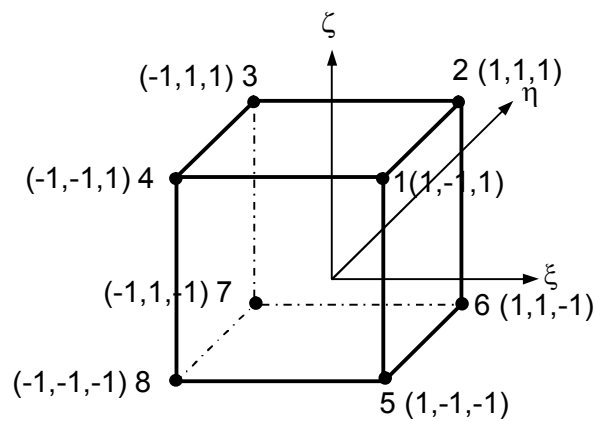


Figure 1.11 : Élément fini tridimensionnel rapporté au système de coordonnées naturelles

Les fonctions de forme correspondantes seront :

$$\begin{aligned}
 N_1 &= \frac{1}{8}(1 - \xi)(1 - \eta)(1 - \zeta) \\
 N_2 &= \frac{1}{8}(1 + \xi)(1 - \eta)(1 - \zeta) \\
 N_3 &= \frac{1}{8}(1 + \xi)(1 + \eta)(1 - \zeta) \\
 N_4 &= \frac{1}{8}(1 - \xi)(1 + \eta)(1 - \zeta) \\
 N_5 &= \frac{1}{8}(1 - \xi)(1 - \eta)(1 + \zeta) \\
 N_6 &= \frac{1}{8}(1 + \xi)(1 - \eta)(1 + \zeta) \\
 N_7 &= \frac{1}{8}(1 + \xi)(1 + \eta)(1 + \zeta) \\
 N_8 &= \frac{1}{8}(1 - \xi)(1 + \eta)(1 + \zeta)
 \end{aligned} \tag{1.31}$$

Les fonctions de forme N_i vérifient les propriétés de la MEF : elles sont nulles dans n'importe quel point exceptant le nœud i et égales à l'unité dans le nœud respectif.

1.12.1 Approximation des déplacements

Le vecteur des déplacements nodaux pour un élément a 24 composantes et peut être écrit sous forme de :

$$\{\delta_e\} = \begin{Bmatrix} \{U_1\} \\ \{U_2\} \\ \vdots \\ \{U_8\} \end{Bmatrix} \quad \text{où } \{U_i\} = \begin{Bmatrix} U_{xi} \\ U_{yi} \\ U_{zi} \end{Bmatrix}, \quad i = 1 \dots 8 \tag{1.32}$$

Où U_{xi} représente le déplacement du nœud i sur la direction de l'axe OX etc.

Le vecteur des déplacements dans l'intérieur de l'élément (vecteur du champ de déplacements) est approximé à l'aide des fonctions de forme N_i $i = 1 \dots 8$:

$$\left. \begin{aligned}
 u_x(x, y, z) &\approx N_1 u_{x1} + N_2 u_{x2} + \dots + N_8 u_{x8} \\
 u_y(x, y, z) &\approx N_1 u_{y1} + N_2 u_{y2} + \dots + N_8 u_{y8} \\
 u_z(x, y, z) &\approx N_1 u_{z1} + N_2 u_{z2} + \dots + N_8 u_{z8}
 \end{aligned} \right\} \tag{1.33}$$

Une autre forme dont la relation (1.31) peut être écrite sera:

$$\begin{Bmatrix} U_x(x, y, z) \\ U_y(x, y, z) \\ U_z(x, y, z) \end{Bmatrix} = \begin{bmatrix} N_1 & 0 & 0 & N_2 & 0 & 0 & \dots & N_8 & 0 & 0 \\ 0 & N_1 & 0 & 0 & N_2 & 0 & \dots & 0 & N_8 & 0 \\ 0 & 0 & N_1 & 0 & 0 & N_2 & \dots & 0 & 0 & N_8 \end{bmatrix} \times \begin{Bmatrix} U_{x1} \\ U_{y1} \\ U_{z1} \\ \vdots \\ U_{x8} \\ U_{y8} \\ U_{z8} \end{Bmatrix} \quad (1.34)$$

Sous une forme condensée, la relation (1.32), s'écrit sous la forme :

$$\{\Delta\} = \begin{Bmatrix} U_x(x, y, z) \\ U_y(x, y, z) \\ U_z(x, y, z) \end{Bmatrix} = [[\Phi_1] \quad [\Phi_2] \quad \dots \quad [\Phi_8]] \{\delta_e\} \quad (1.35)$$

Où $\{\Delta\}$ est le vecteur du champ de déplacement.

1.12.2. Approximation de la géométrie

L'élément fini en cause est isoparamétrique, donc aussi bien sa géométrie que l'approximation des déplacements intérieurs sont obtenus à l'aide des mêmes fonctions de forme, comme il en suit :

$$x = \sum_{i=1}^8 N_i x_i \quad ; \quad y = \sum_{i=1}^8 N_i y_i \quad ; \quad z = \sum_{i=1}^8 N_i z_i \quad (1.36)$$

Ou bien :

$$x = [N_1 \quad N_2 \quad N_3 \quad \dots \quad N_8] \times \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_8 \end{Bmatrix} = [N] \times \{X\} \quad (1.37)$$

De même, $y = [N] \cdot \{Y\}$ et $z = [N] \cdot \{Z\}$.

$[N] = [N_1 \quad N_2 \quad \dots \quad N_8]$ = matrice ligne des fonctions de forme ;

$\{X\}, \{Y\}, \{Z\}$ = vecteur des coordonnées nodales de l'élément.

1.12.3. Déduction de la matrice [B]

Les relations entre les déplacements et les déformations relatifs sont :

$$\left. \begin{aligned} \varepsilon_x &= \frac{\partial u_x}{\partial x} = \frac{\partial u}{\partial x} \quad ; \quad \gamma_{xy} = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \\ \varepsilon_y &= \frac{\partial u_y}{\partial y} = \frac{\partial v}{\partial y} \quad ; \quad \gamma_{yz} = \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \\ \varepsilon_z &= \frac{\partial u_z}{\partial z} = \frac{\partial w}{\partial z} \quad ; \quad \gamma_{zx} = \frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \end{aligned} \right\} \quad (1.38)$$

$$\{\varepsilon\} = \begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \\ \gamma_{yz} \\ \gamma_{zx} \\ \gamma_{xy} \end{Bmatrix} = \begin{bmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial v}{\partial y} \\ \frac{\partial w}{\partial z} \\ \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \\ \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \\ \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \end{bmatrix} = [\partial] \begin{Bmatrix} u \\ v \\ w \end{Bmatrix} = [\partial]\{\Delta\} \quad (1.39)$$

$$\text{En introduisant la valeur de } \{\Delta\} = [[\Phi_1] \quad [\Phi_2] \quad \dots \quad [\Phi_8]]\{\delta_e\} \quad (1.40)$$

On obtient :

$$\{\varepsilon\} = \underbrace{\begin{bmatrix} \frac{\partial N_1}{\partial x} & 0 & 0 & \frac{\partial N_2}{\partial x} & 0 & 0 & \dots & \frac{\partial N_8}{\partial x} & 0 & 0 \\ 0 & \frac{\partial N_1}{\partial y} & 0 & 0 & \frac{\partial N_1}{\partial y} & 0 & \dots & 0 & \frac{\partial N_1}{\partial y} & 0 \\ 0 & 0 & \frac{\partial N_1}{\partial z} & 0 & 0 & \frac{\partial N_1}{\partial z} & \dots & 0 & 0 & \frac{\partial N_1}{\partial z} \\ 0 & \frac{\partial N_1}{\partial z} & \frac{\partial N_1}{\partial y} & 0 & \frac{\partial N_2}{\partial z} & \frac{\partial N_2}{\partial y} & \dots & 0 & \frac{\partial N_8}{\partial z} & \frac{\partial N_8}{\partial y} \\ \frac{\partial N_1}{\partial z} & 0 & \frac{\partial N_1}{\partial x} & \frac{\partial N_2}{\partial z} & 0 & \frac{\partial N_2}{\partial x} & \dots & \frac{\partial N_8}{\partial z} & 0 & \frac{\partial N_8}{\partial x} \\ \frac{\partial N_1}{\partial y} & \frac{\partial N_1}{\partial x} & 0 & \frac{\partial N_2}{\partial y} & \frac{\partial N_2}{\partial x} & 0 & \dots & \frac{\partial N_8}{\partial y} & \frac{\partial N_8}{\partial x} & 0 \end{bmatrix}}_{[B]} \cdot \{\delta_e\} \quad (1.41)$$

Ou, sous forme condensée :

$$\{\varepsilon\} = [[B_1] \quad [B_1] \quad \dots \quad [B_1]] \cdot \{\delta_e\} \quad (1.42)$$

1.12.4. Détermination de la forme finale de la matrice de rigidité de l'élément :

En connaissant les matrices $[B]$ et $[H]$, on peut déterminer l'expression la matrice de rigidité pour l'élément fini en cause [3].

Pour un élément de volume parallélépipédique, le volume élémentaire est :

$$dV = dx dy dz.$$

$$[k^{(e)}] = \int_V [B]^T \cdot [H][B] dV = \int_{-1}^1 \int_{-1}^1 \int_{-1}^1 [B]^T \cdot [H] \cdot [B] \cdot \det[J] d\xi d\eta d\zeta \quad (1.43)$$

L'utilisation du système naturel de coordonnées présente l'avantage qu'il permet le calcul numérique des intégrales avec une très bonne précision.

$$[H] = a \begin{bmatrix} 1 - \nu & \nu & \nu & 0 & 0 & 0 \\ \nu & 1 - \nu & \nu & 0 & 0 & 0 \\ \nu & \nu & 1 - \nu & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1-2\nu}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1-2\nu}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1-2\nu}{2} \end{bmatrix} \quad (1.44)$$

$$a = \frac{E}{(1 + \nu) + (1 - 2\nu)}$$

CHAPITRE 2

Méthodes des éléments finis

2.1. Introduction

L'avenir est à l'outil numérique et à la réalité virtuelle; la simulation et l'analyse numérique se sont développées ces dernières années dans la recherche industrielle privée ou publique (constructeurs automobile, aéronautique, spatial...) améliorant ainsi la productivité des industriels et la vie de tous les consommateurs [4].

La solution d'analyse par éléments finis optimise le prototypage virtuel et la simulation numérique pour l'entreprise en utilisant les hypothèses de la mécanique de la rupture en élasticité et de la dynamique. Ainsi elle simule et améliore les performances.

La simulation et l'analyse numérique ont un impact direct sur la qualité des composants, des matériaux et des produits sujets à de fortes exigences opérationnelles (sécurité, endommagement...). et le temps de mise sur le marché des produits ; de plus, elles accroissent les performances des solutions de gestion du cycle de vie du produit. Elles sont apparues, dans le désir de minimiser le coût d'une étude avec un nombre d'essais optimisés. La difficulté est de créer un modèle C.A.O. se rapprochant au maximum de la réalité (matériaux, conditions aux limites...). L'étude ne sera validée qu'à partir d'essais physiques entraînant des mesures pour la comparaison entre la réalité et la simulation (crash test...).

D'un point de vue technique, les résultats obtenus grâce à ABAQUS sont le bilan des énergies, des forces nodales, des déformations, des déplacements, des contraintes, des vitesses, des accélérations et de toutes les grandeurs physiques nécessaires à la conception d'un modèle.

ABAQUS est un code de calcul par la méthode des éléments finis créé en 1978. Il est avant tout un logiciel de simulation de problèmes très variés en mécanique. Simuler la réponse physique des structures soumises à des chargements, des températures, des impacts ou autres conditions extérieures, Il est connu et répandu, en particulier pour ses traitements performants de problèmes non-linéaires [5].

2.2. Généralité sur le code de calcul ABAQUS

ABAQUS compose de trois produits : ABAQUS/Standard, ABAQUS/Explicit et ABAQUS/CAE, figure 2.1.

2.2.1. ABAQUS/Standard

ABAQUS Est un solveur généraliste qui recourt à un schéma traditionnel d'intégration implicite [6].

- Résolution basée sur l'algorithme de Newton-Raphson et la méthode de Riks
- Problèmes linéaires et non linéaires
- Géométrie 1D, 2D, 3D et Axisymétrique
- Nombreuses procédures d'analyse dans le domaine temporel et fréquentiel.

2.2.2. ABAQUS/Explicit

ABAQUS/ Explicite Un schéma d'intégration explicite pour résoudre des problèmes dynamique ou quasi-statiques non-linéaires.

- Analyse non linéaire (discrétisation spatiale et temporelle)
- Problème transitoire et dynamique des structures
- Résolution basée sur la méthode explicite d'intégration en temps

2.2.3. ABAQUS/CAE

ABAQUS /CAE Constitue une interface intégrée de visualisation et de modélisation pour lesdits solveurs. Chacun de ces produits est complété par des modules additionnels et optionnels, spécifiques à certaines applications.

ABAQUS fut d'abord conçu pour analyser les comportements non-linéaires. Il possède en conséquence une vaste gamme de modèles de matériau. Ses modélisations d'élastomères, en particulier, méritent d'être reconnues.

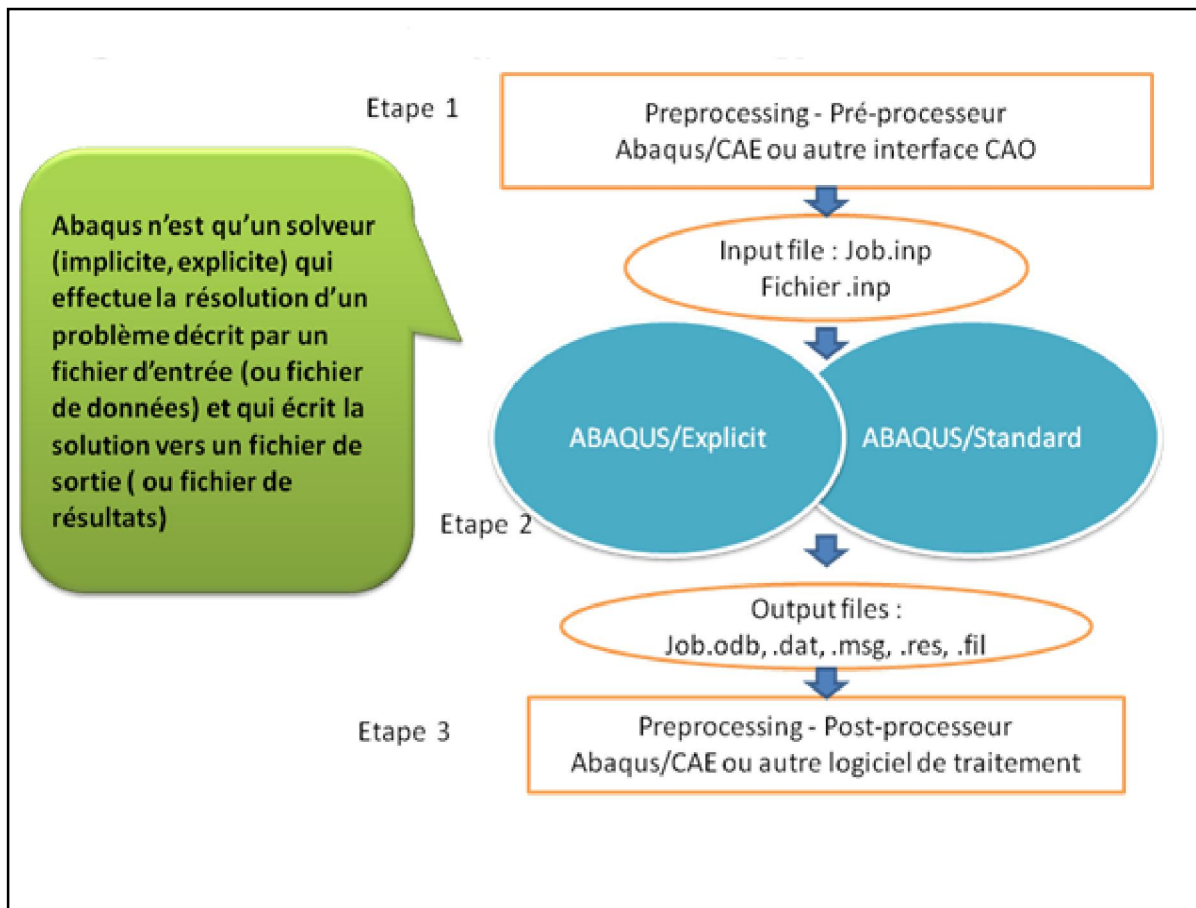


Figure 2.1 : Structure d'une analyse sous Abaqus [6]

2.3. Les différents fichiers pour la simulation sous Abaqus

2.3.1. Fichier d'entrée (Input file)

2.3.1.1. Extension .inp : contient des mots clés décrivant la géométrie, les matériaux, les conditions limites et les chargements de la structure étudiée

➤ **Génération du fichier .inp :**

- Avec un éditeur (fichier texte) ou programme en connaissant les divers mots clés de commande Abaqus
- Avec l'interface graphique Abaqus/CAE
- Avec un logiciel annexe (I-DEAS, NASTRAN, ...)

2.3.2. Fichier de résultats (output file)

2.3.2.1. Extension **.odb** : Contient les contours et courbes de résultats

2.4. Autres fichiers créés par Abaqus lors de la simulation

- Fichier **.com** : exécutable de vos calculs
- Fichier **.res** : fichier binaire de reprise
- Fichier **.dat** : liste des résultats, résumé de votre modélisation
- Fichier **.msg** : résumé du déroulement du calcul en cours, message d'erreur
- Fichier **.fil** : fichier binaire des résultats
- Fichier **.log** : fichier de procédure
- Fichier **.sta** : Fichier statistique ... ets.

2.5. Organisation de l'interface Abaqus CAE (complète Abaqus environnement)

Il est important de savoir que le module CAE figure 2.2 que vous allez utiliser pour créer votre jeu de données est relativement récent (quelques années) et qu'il est plutôt destiné à la mise en données de problèmes relativement classiques.

Donc pas toutes les commandes utilisables dans Abaqus et fait souvent le choix de paramètres par défaut [7].

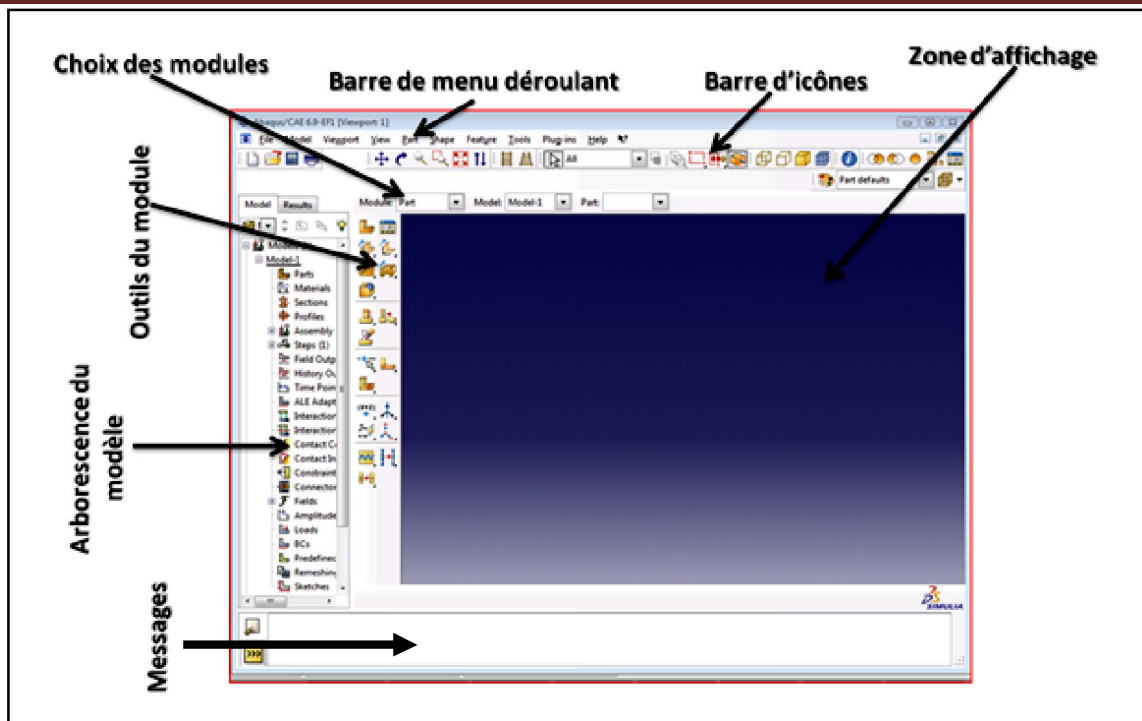


Figure 2.2 : Organisation d'abaqus [7]

2.6. Les modules

La réalisation complète d'un jeu de données s'effectue après un passage successif dans les modules, figure 2.3 [8].

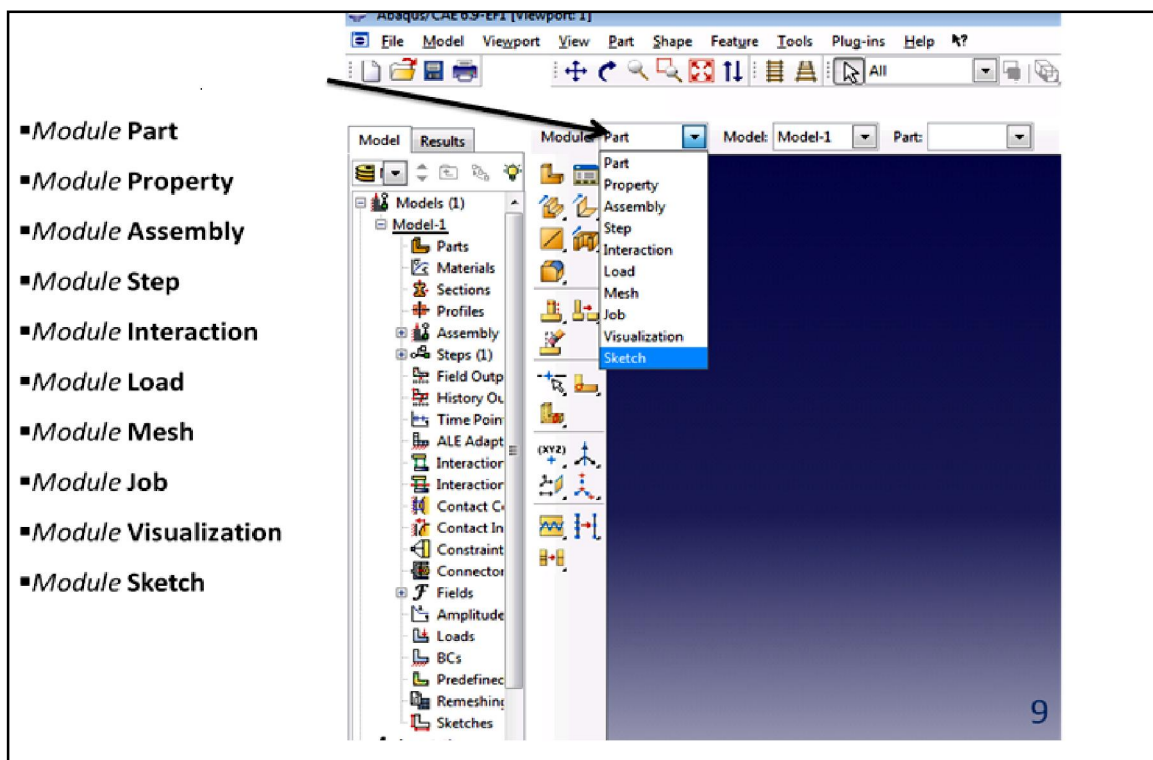


Figure 2.3 : Réalisation de la mise en données d'un problème

2.6.1. Module part

PART ce module qu'est définie la géométrie des entités du modèle. Figure 2.4 et figure 2.5. Elles comprennent à la fois les parties déformables et les parties rigides. Pour créer les créer, il faut utiliser l'icône de création en haut à gauche.

Il ne vous reste plus qu'à sélectionner convenablement :

- L'espace de modélisation : 3D par exemple
- Le type d'entité à générer : déformable
- La forme de l'entité.

Dans tous les cas vous passez dans un autre environnement de type CAO où vous dessinez vos pièces.

Dans le cas d'un solide rigide, il est nécessaire de définir un point de référence. C'est en fait en ce point que sont définis les degrés de liberté du solide sur lesquels vous imposerez des conditions aux limites par la suite. Ceci se fait dans le menu Tools, grâce à la commande Référence point.

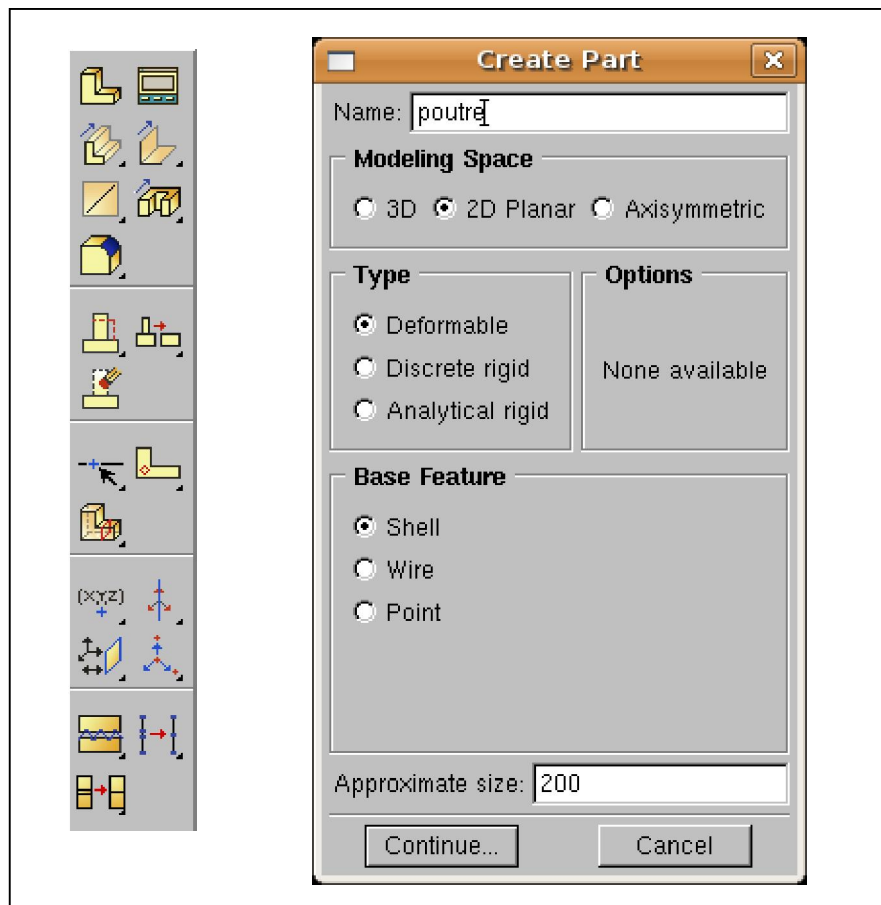


Figure 2.4 : Module Part

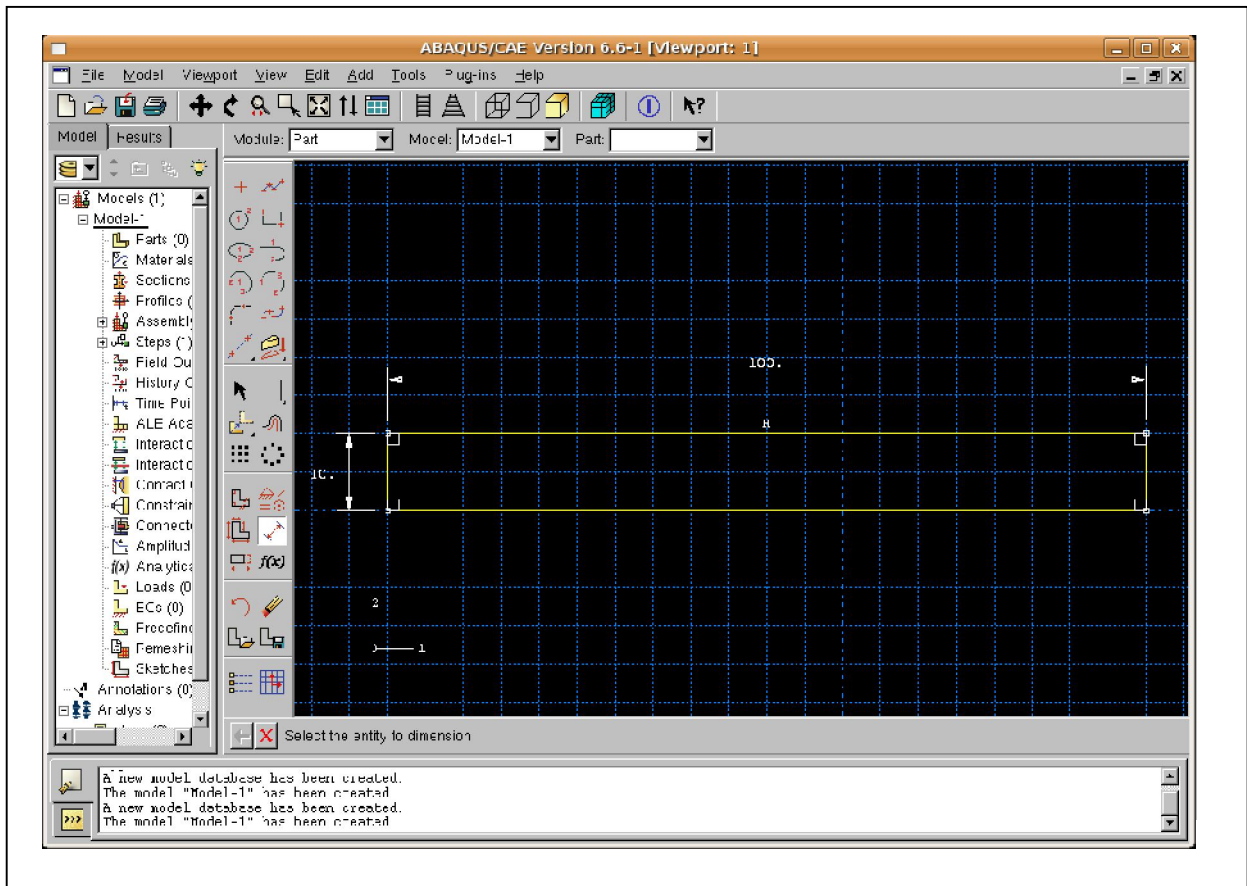


Figure 2.5 : création de la géométrie [9]

2.6.2. Module Property

On y définit les propriétés du ou des matériaux figure 2.6 (des parties déformables du modèle) et la masse des corps rigides si nécessaire. Dans le cas présent, vous avez à définir des propriétés élastiques et/ou plastiques. Pour créer un matériau, il suffit de cliquer sur l'icône en haut à gauche.

Il vaut mieux ensuite donner un nom spécifique à chacun de vos matériaux ce qui permet éventuellement la création d'une base de données, il vous reste à définir :

- Les propriétés élastiques via Mechanical, Elasticity, Elastic (module d'Young et le coefficient de Poisson pour un matériau isotrope).
- Les propriétés plastiques via Mechanical, Plasticity, Plastic. (Sous l'hypothèse d'un écrouissage isotrope, la courbe d'écrouissage est entrée sous la forme : limite d'élasticité/déformation plastique. Dans cette courbe,

la contrainte est de type Cauchy (F divisée par la surface actuelle) et la déformation est logarithmique ($\ln(L/L_0)$).

Il faut ensuite affecter les propriétés matériau à une entité du maillage : pour cela il est nécessaire de créer tout d'abord une section avec l'icône en forme de poutre. Vous aurez à faire référence à un matériau et il ne vous restera plus qu'à Donner une valeur à l'épaisseur.

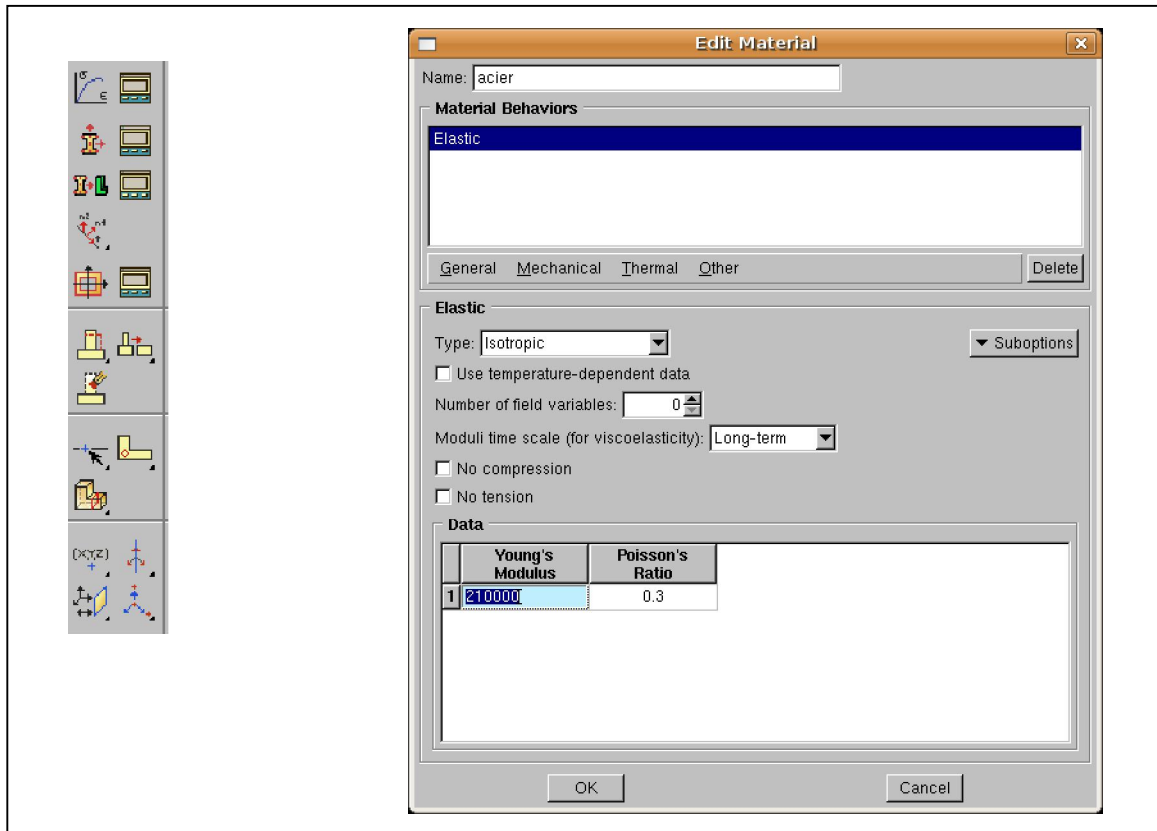


Figure 2.6 : propriété Matériaux et section

2.6.3. Module Assembly

Assembly ce module, il s'agit de créer une instance figure 2.7 à partir des entités que vous avez définies dans le module Part. Ces instances seront celles que vous manipulerez par la suite dans le modèle. Dans le cas général, à partir de l'icône en haut à gauche, pour créer une instance vous sélectionnez tous les éléments créés dans Part. Vous pouvez le cas échéant faire subir des mouvements d'ensemble à ces instances pour les repositionner dans l'espace.

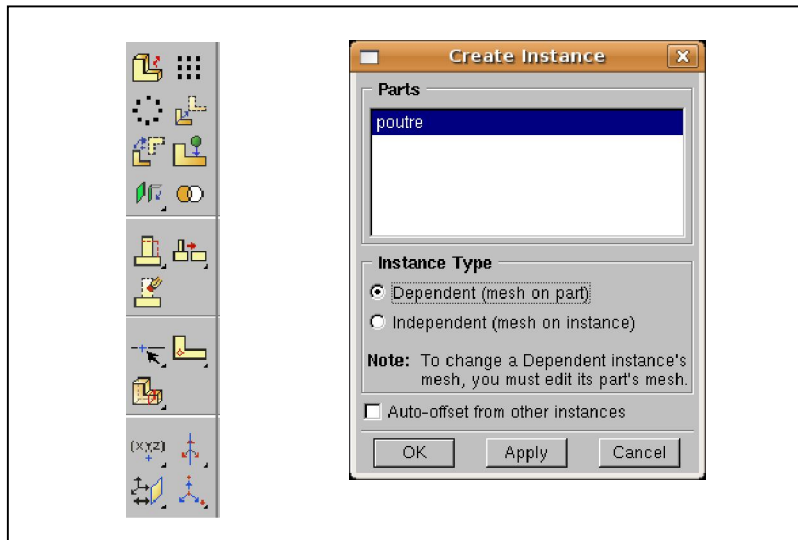


Figure 2.7 : Instance et assemblage

2.6.4. Module Step

C'est ici que vous choisirez, figure 2.8 dans le cas d'études de crash ou d'impact, le module « Dynamic explicit ». Vous remarquerez au passage les nombreuses autres possibilités qui sont offertes par ce code de calcul.

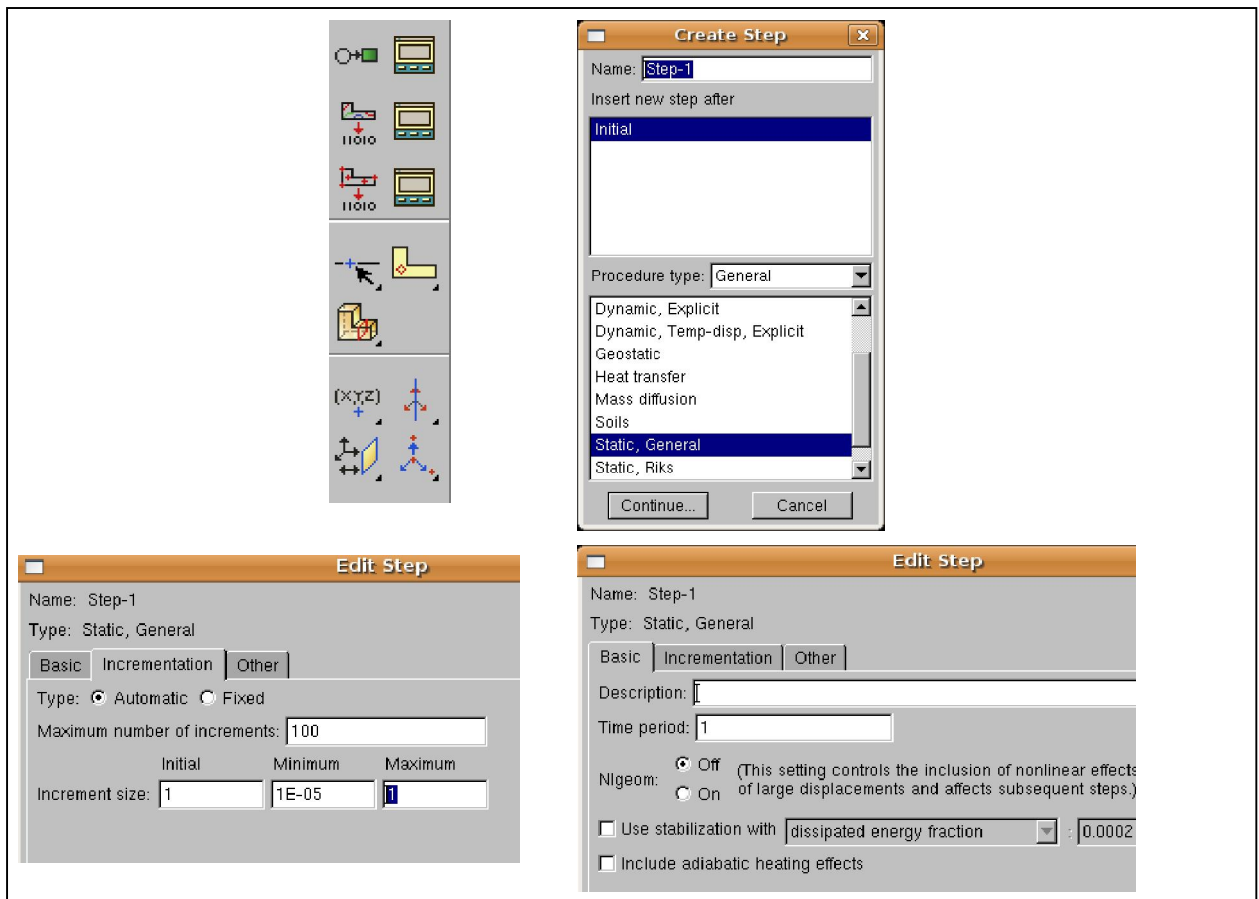


Figure 2.8 : Création de step

2.6.5. Module Interaction

Grâce ce module, il est possible de spécifier toutes les interactions entre les différentes parties et régions du modèle, qu'elles soient mécaniques, thermiques ou autres. Il faut savoir qu'ABAQUS ne prend en compte que les interactions explicitement définies, la proximité géométrique n'étant pas suffisante Figure 2.9.

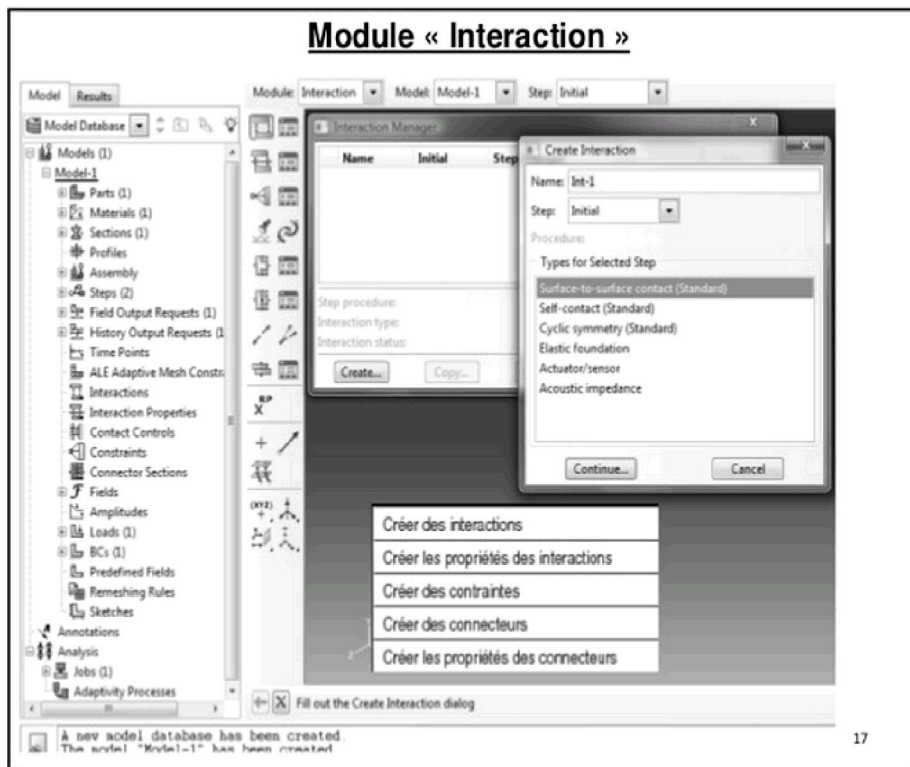


Figure 2.9 : interaction

2.6.6. Module Load

Le module Load permet de spécifier tous les chargements, conditions limites et champs. Figure 2.10 .Il faut savoir que les chargements et le conditions limites sont dépendants des steps, par exemple une force est appliquée au step 1 mais inactive au step 2.

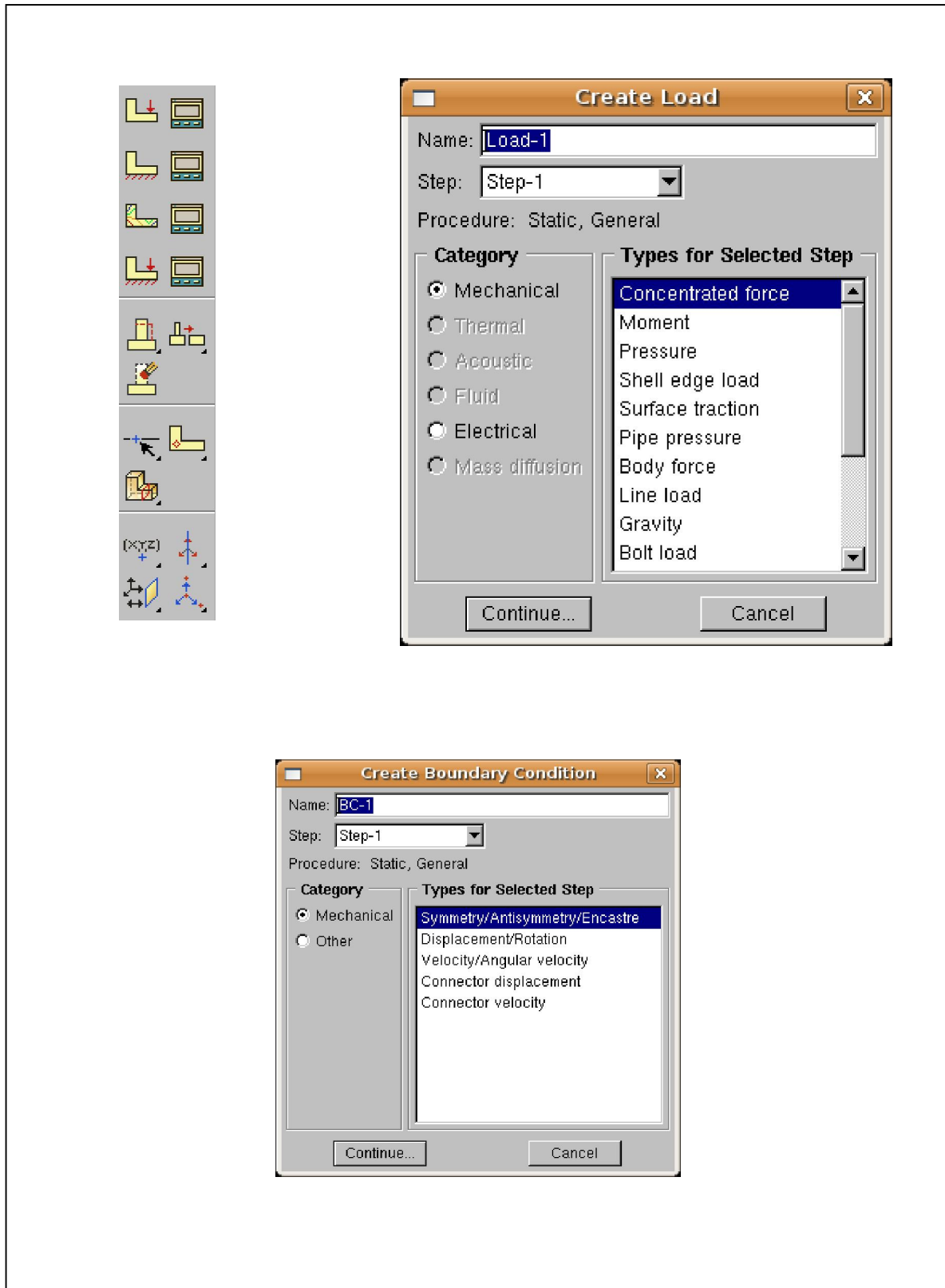


Figure 2.10 : Création des chargements

2.6.7. Module Mesh

Mesh contient tous les outils nécessaires pour générer un maillage élément fini sur un assemblage, figure 2.11.

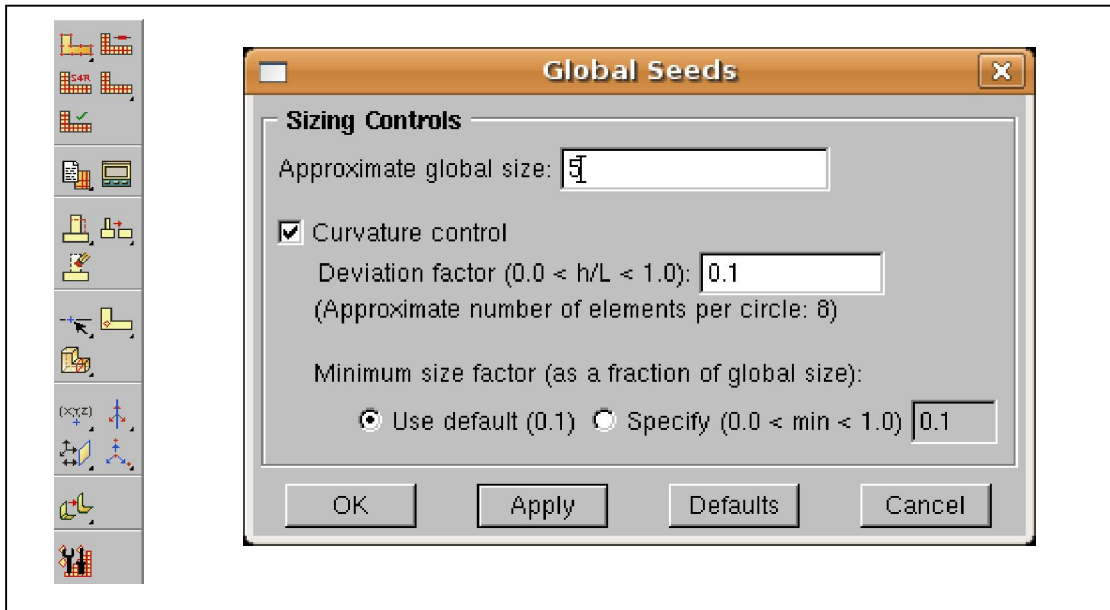


Figure 2.11 : Création de maillage

2.6.8. Module Job

Le « lancement » du calcul est classique. Figure 2.12, Après avoir lancé le calcul, vous pouvez suivre l'évolution du procès en cliquant sur Monitor : vous pouvez ainsi avoir en temps réel l'évolution des incréments de chargement et apprécier la convergence des calculs.

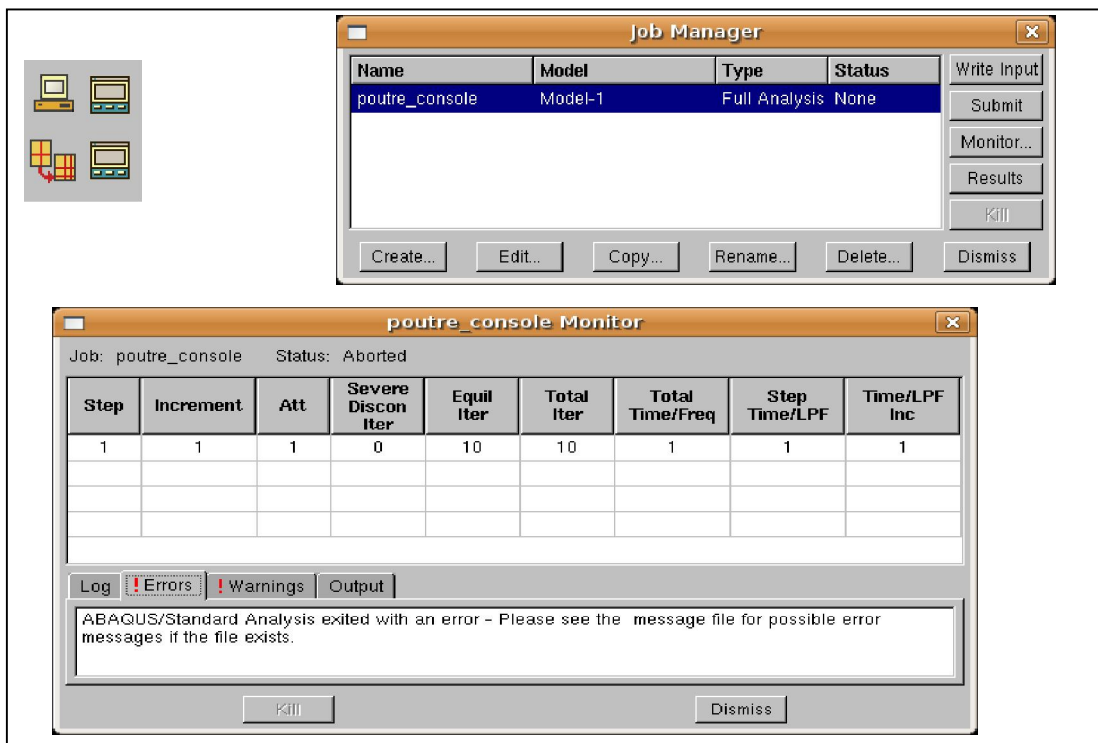


Figure 2.12 : Lancement de calcul

2.6.9. Module visualization

Visualization Est un module qui affiche tout les résultats de la pièce soit contrainte ou déformation, figure 2.13.

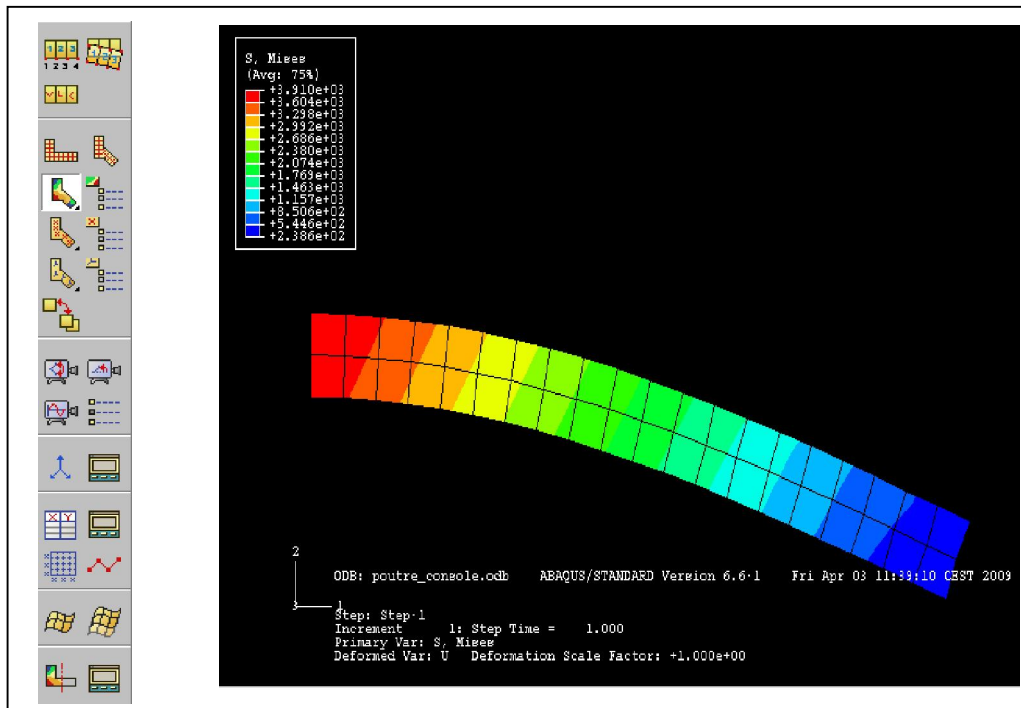


Figure 2.13 : Affichage des résultats [9]

2.6.10. Module sketch

Sketch permet de créer des formes bidimensionnelles qui ne sont pas associés à un objet. Il peut être utilisé pour des extrusions par exemple.

2.7. L'arbre du modèle

L'arbre donne une description visuelle de la hiérarchie des objets du modèle. Tous les objets sont indiqués par de petites icônes avec en parenthèses le nombre de ces objets. L'ordre de l'arbre reflète l'ordre classique d'élaboration du modèle.

A partir de cet arbre il est possible de retrouver la plupart des fonctionnalités de la barre de menu principale. Par exemple, double cliquer sur Part permet de créer une nouvelle Part comme sélectionner **Part** -> **Create** dans la barre de menu.

2.8. Les éléments dans ABAQUS

- Un très large choix d'éléments (plus de 100), figure 2.14.
- Chaque élément est caractérisé par : famille, ddl, nombre de nœuds, intégration.
- Possibilités de programmer de nouveaux éléments (en FORTRAN, sur ABAQUS/Standard) [10].

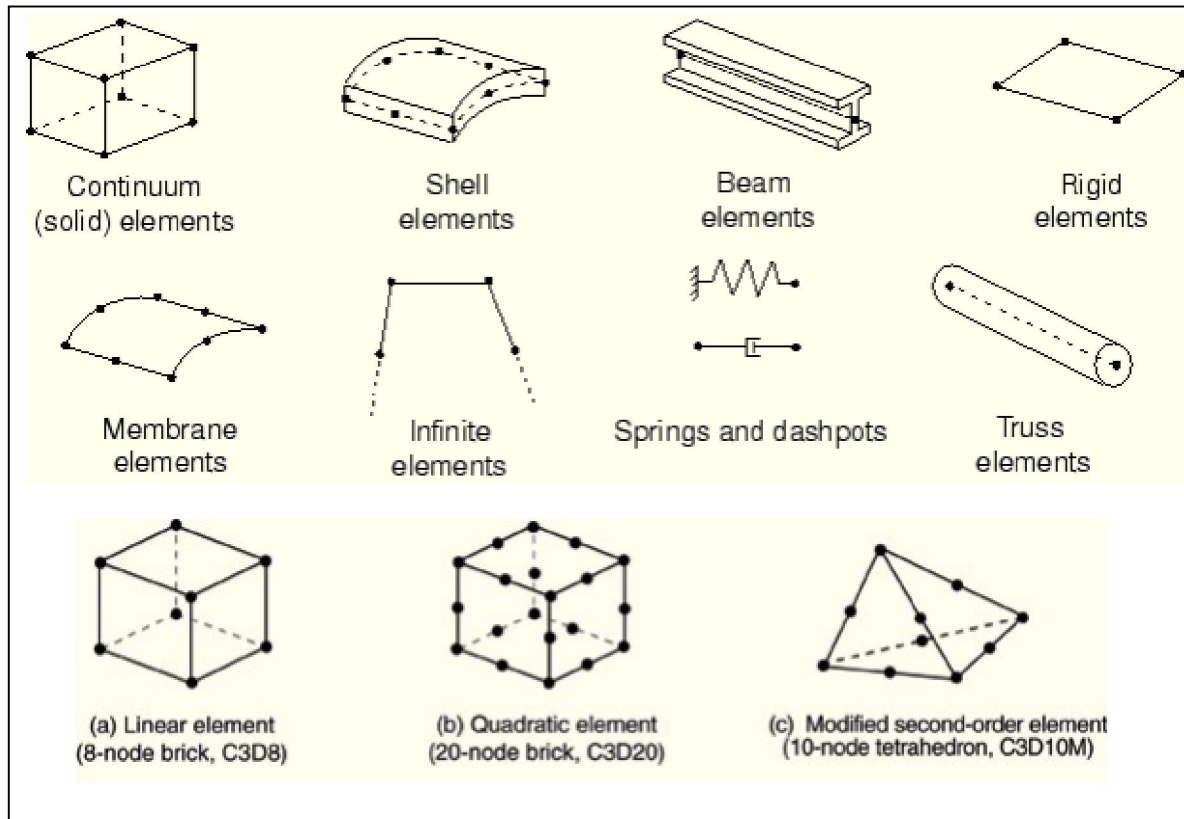


Figure 2.14 : Quelques éléments dans Abaqus [10]

2.8.1. Degré de liberté (DDL)

- 1 Translation en direction 1
- 2 Translation en direction 2
- 3 Translation en direction 3
- 4 Rotation autour de l'axe 1
- 5 Rotation autour de l'axe 2
- 6 Rotation autour de l'axe 3
- 7 Voilement dans les éléments poutres à profil ouvert
- 8 Pression acoustique, pression de pores, ou pression hydrostatique
- 9 Potentiel électrique
- 11, 12+ Température k

Les éléments axisymétriques ont exceptionnellement, les degrés de liberté de déplacement et de rotation suivant :

- 1 Translation en direction r
- 2 Translation en direction z
- 6 Rotation autour du plan r - z

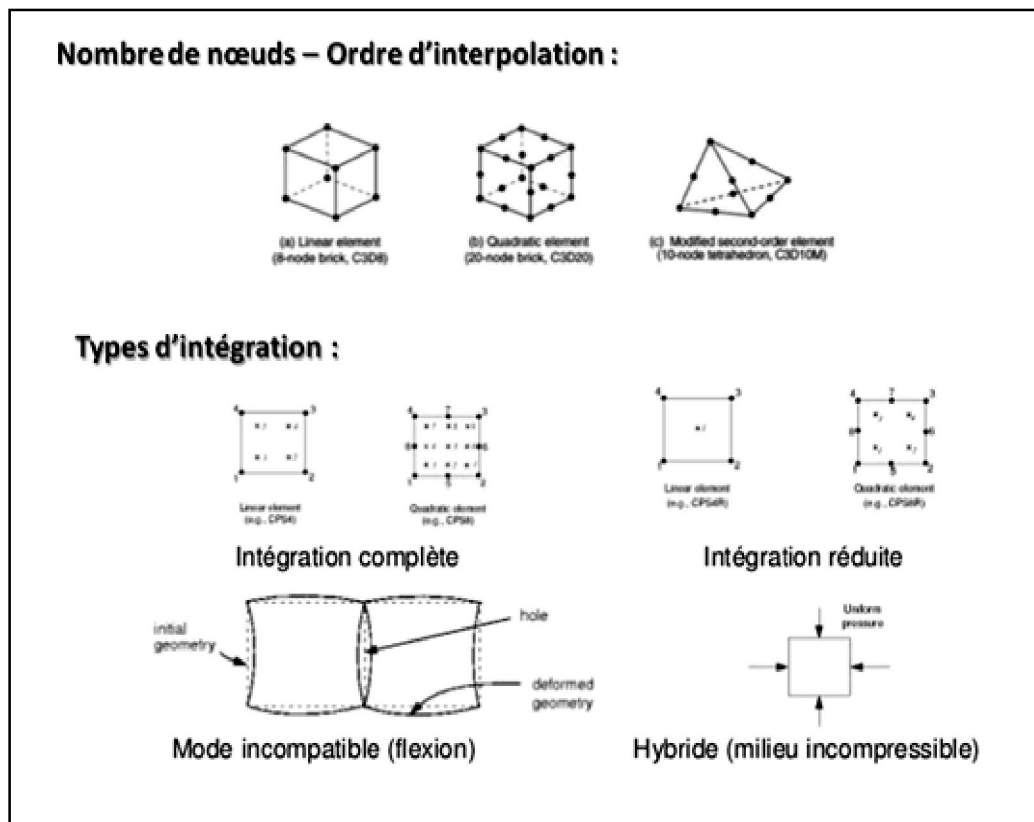


Figure 2.15 : Les types d’intégrations et le nombre de nœuds

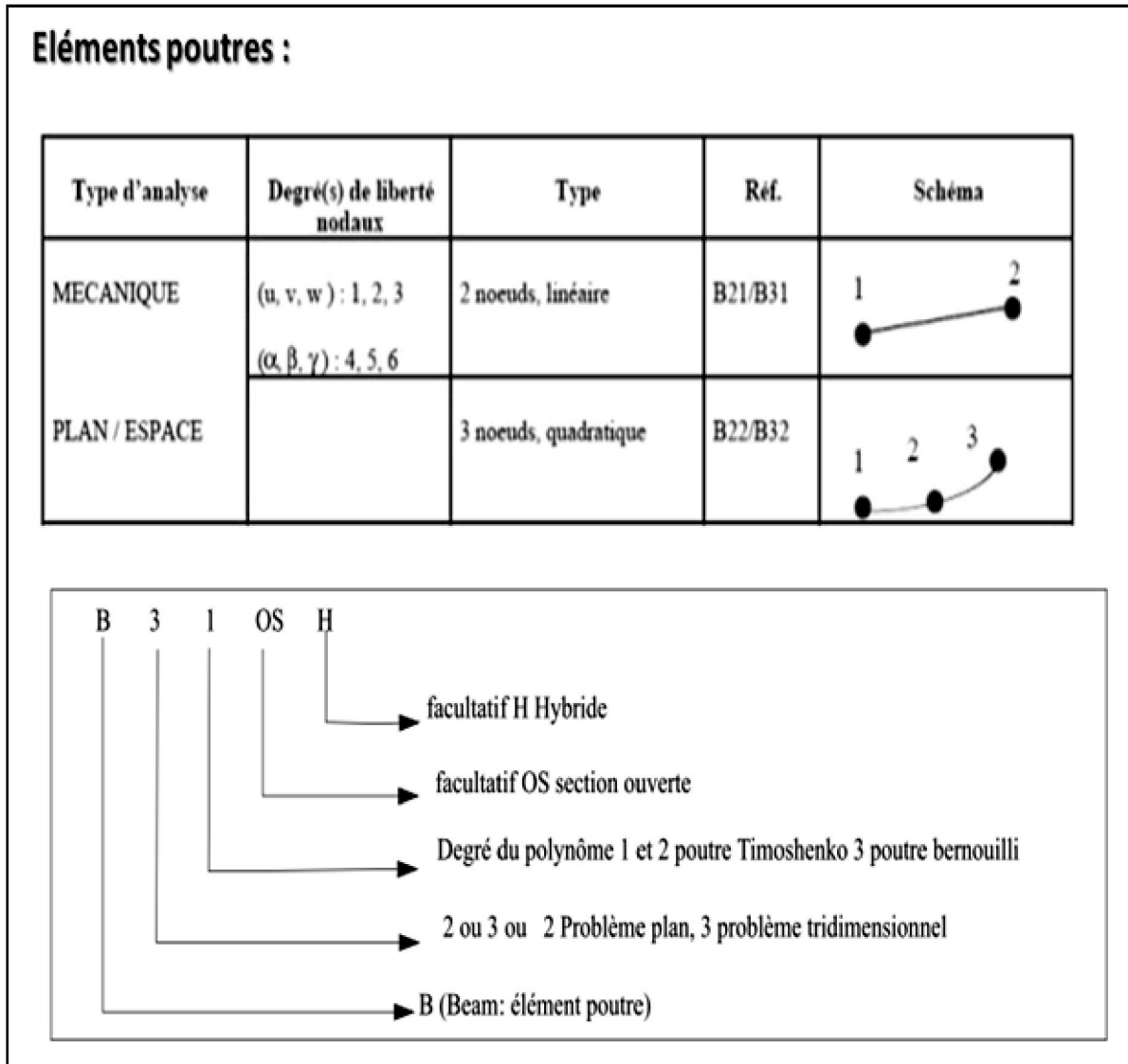


Figure 2.16. Exemple d'élément poutre






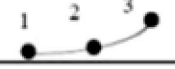
Éléments solides 1D				
Type d'analyse	Degré(s) de liberté nodaux	Type	Réf.	Schéma
MECANIQUE	(u, v, w) : 1, 2, 3	2 noeuds, linéaire	C1D2	
		3 noeuds, quadratique	C1D3	
TRANSFERT DE CHALEUR	(température) : 11	2 noeuds, linéaire	DC1D2	
		3 noeuds, quadratique	DC1D3	
TEMPERATURE-DEPLACEMENT	(u, v, w) : 1, 2, 3 (température) : 11	2 noeuds, linéaire	DC1D2T	
		3 noeuds, quadratique	DC1D3T	

Figure 2.17 : Éléments solides 1D









Éléments solides 2D :				
Type d'analyse	Degré(s) de liberté nodaux	Type	Réf.	Schéma
MECANIQUE	(u, v) : 1, 2	3 noeuds, linéaire	CPE3 CPS3	
DEFORMATIONS PLAINES / CONTRAINTES PLAINES		6 noeuds, quadratique	CPE6 CPS6	
	(u, v) : 1, 2	4 noeuds, linéaire	CPE4 CPS4	
		8 noeuds, quadratique	CPE8 CPS8	
TRANSFERT DE CHALEUR	(température) : 11	3 noeuds, linéaire	DC2D3	
		4 noeuds, linéaire	DC2D4	
TEMPERATURE-DEPLACEMENT	(u, v) : 1, 2 (température) : 11	4 noeuds, linéaire	CPE4T CPS4T	
		8 noeuds, déplacement quadratique, température linéaire	CPE8T CPS8T	

Figure 2.18 : Éléments solides 2D

Éléments axisymétriques 2D :

Type d'analyse	Degré(s) de liberté nodaux	Type	Ref.	Schema
MECANIQUE	(u, v) : 1, 2	3 nœuds, linéaire	CAX3	
		6 nœuds, quadratique	CAX6	
	(u, v) : 1, 2	4 nœuds, linéaire	CAX4	
		8 nœuds, quadratique	CCAX8	
TRANSFERT CHALEUR	DE (température) : 11	3 nœuds, linéaire	DCAX3	
		4 nœuds, linéaire	DCAX4	
TEMPÉRATURE-DEPLACEMENT	(u, v) : 1, 2 (température) : 11	4 nœuds, linéaire	CAX4T	
		8 nœuds, quadratique, déplacement, température, linéaire	CAX8T	

Figure 2.19 : Éléments axisymétriques 2D

Éléments coques (Shell) :

Type d'analyse	Degré(s) de liberté nodaux	Type	Ref.	Schema
MECANIQUE	(u, v, w) : 1, 2, 3 ($\theta_x, \theta_y, \theta_z$) : 4, 5, 6	3 nœuds, linéaire	S3R	
		4 nœuds, linéaire	S4R	
		8 nœuds, quadratique	S8R	
TRANSFERT CHALEUR	DE (température) : 11, 12, ... 17	4 nœuds, quadratique	DS4	
TEMPÉRATURE-DEPLACEMENT COUPLEE	(u, v, w) : 1, 2, 3 ($\theta_x, \theta_y, \theta_z$) : 4, 5, 6 (température) : 11, 12, ... 17	8 nœuds, quadratique	S8RT	

Éléments 3D :

Tetrahedra

Triangular prisms (wedges)

Hexahedra

La dénomination des éléments est la suivante :

S 3 R 5 W

- Gauchissement pris en compte
- Option: 5 = S4R ou S formulation en petite déformation
- Option R = intégration réduite
- Nombre de nœuds
- S élément standard, SC Continuum shell, STRI élément mince

Les éléments « général purpose » qui effectivement seront souvent les plus utilisés sont les suivants :

- S3 S3R éléments triangulaires
- S4 S4R S4RS

Figure 2.20 : Éléments coques et 3D

2.9. Les Types de non-linéarités

- Matériaux
- Géométriques
- Contact

2.9.1. Non-linéarités matérielles

- Hyper-élasticité
- Plasticité
- Visco-plasticité
- Endommagement
- Non-isotrope (y compris laminates = composites)

2.9.2. Non-linéarités géométriques

- grandes déformations
- Grands déplacements
- Grandes rotations
- Instabilités (bifurcations, points-limites)

2.9.3. Non-linéarités de contact

- Grands déplacements
- Contact 2D et 3D
- Rigide/Rigide, Rigide/Déformable, Déform./Déform.
- Lois de frottement diverses et complexes

2.9. Domaines physiques

- Mécanique
- Thermique
- Électrique (piézo et thermique)
- Problèmes couplés
- Statique et dynamique
- Linéaires et non linéaires

CHAPITRE 3

Formulation non linéaire géométrique

3.1. Introduction

Non-linéaire géométrique (on dit parfois de manière équivalente "en grands déplacements"), signifie que l'on abandonne l'hypothèse que la configuration courante peut-être confondue avec la configuration initiale. Cela a pour conséquence que la configuration sur laquelle doivent être calculés les termes des travaux virtuels, c'est-à-dire la configuration actuelle, devient une inconnue, ce qui introduit une non-linéarité dans le problème. Pour résoudre le problème, il faut donc partir d'une prédiction de cette configuration inconnue (on prend en général la configuration de départ), puis itérer en corrigeant cette configuration à chaque itération.

La résolution des problèmes non-linéaires nécessite la mise en œuvre d'algorithmes adaptés, mais dont la convergence n'est pas forcément garantie. Dès qu'on se lance dans un calcul non-linéaire, on ne peut pas être sûr d'obtenir une solution quelque soit les données et chargements du problème [11].

3.2. La description Lagrangienne

Dans un espace euclidien, on étudie le mouvement d'un corps constitué d'une infinité de points matériels. Sous l'effet de sollicitations mécaniques, les points se déplacent et ce corps se déforme. Nous allons suivre le trajet de l'ensemble des points matériels constitutifs du corps.

L'ensemble des positions qu'occupent ces points à chaque instant s'appelle une configuration. La figure 3.1 représente les trois configurations typiques dans L'analyse non linéaire : C_0 est la configuration initiale connue, C_1 est la configuration Connue et en équilibre au début d'un pas de chargement et C_2 est la configuration Recherchée satisfaisant les conditions d'équilibre.

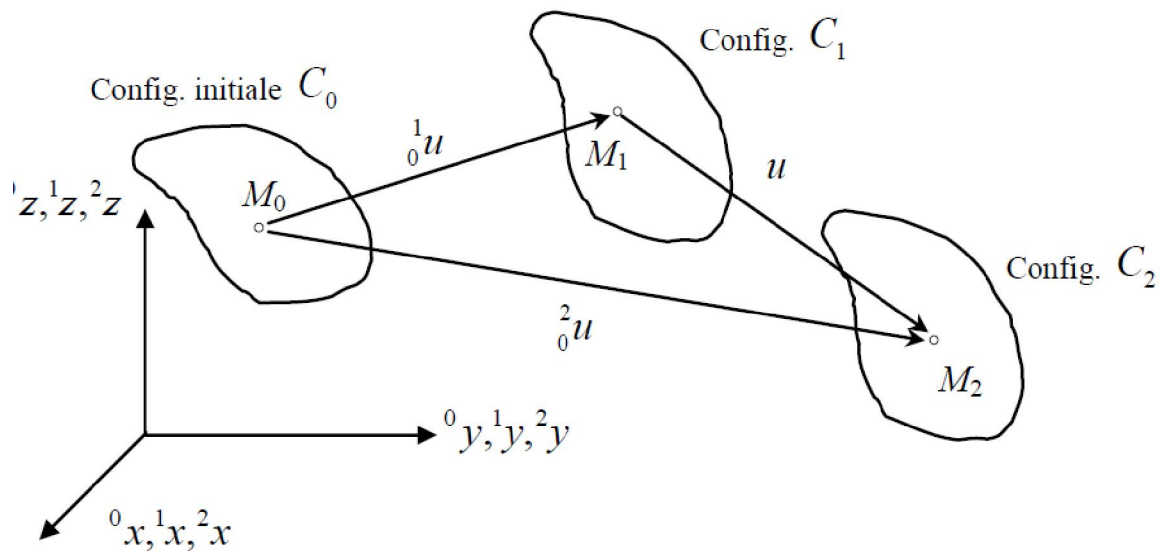


Figure 3.1 : Configurations initiale, intermédiaire et finale d'un corps solide

Si l'on prend pour configuration de référence la configuration initiale C_0 , la description est dite Lagrangienne totale (D.L.T). Si l'on prend pour configuration de référence la configuration courante, la description est dite Lagrangienne actualisée (D.L.A) [11].

3.2.1. La formulation Lagrangienne totale

Où la configuration de référence est représentée par la configuration initiale C_0 . L'attrait pour cette formulation se justifie par deux avantages essentiels : la constance du domaine d'intégration et la facilité de changement des conditions aux limites. Cependant, la présence d'une configuration de référence fixe devient un inconvénient lors du calcul des contraintes.

3.2.2. La formulation Lagrangienne actualisée

Où la configuration de référence est représentée par la dernière position d'équilibre connue du corps. Cette formulation permet de palier à l'inconvénient majeur présenté par la formulation Lagrangienne totale : le calcul des contraintes se fait dans la configuration actuelle, ce qui permet de redonner un sens physique aux valeurs obtenues. Ces contraintes sont confondues avec les contraintes de Piola-Kirchhoff dans la configuration de référence.

Cette formulation nécessite l'actualisation de la configuration de référence à chaque instant. Cette opération peut apparaître comme un inconvénient de la méthode par le surcroît de calcul qu'elle entraîne et la variation permanente des domaines d'intégration.

3.3. Formulation non linéaire géométrique

3.3.1. Principe des travaux virtuels

Les modèles d'éléments finis de type déplacements sont basés sur le principe des travaux virtuels, ou encore des déplacements virtuels. Le principe exige que la somme du travail virtuel externe appliqué sur un corps et le travail virtuel interne stocké dans le corps doit être égale à zéro [12].

$$\delta W = \int_{^2V} {}^2\sigma_{ij} \delta({}^2e_{ij}) d^2V - \delta {}^2R = 0 \quad (3.1)$$

Où $\delta {}^2R$ est le travail virtuel externe des forces volumiques 2f_i appliquées sur le volume 2V

De la configuration et des forces surfaciques 2t_i appliquées sur la surface externe 2S de la configuration C_2 .

Le travail virtuel des forces externes peut être écrit sur la configuration actuelle C_2 sous la forme :

$$\delta {}^2R = \int_{^2V} {}^2f_i \delta u_i d^2V + \int_{^2S} {}^2t_i \delta u_i d^2S \quad (3.2)$$

La configuration actuelle déformée C_2 étant inconnue, l'équation (3.1) est inexploitable sous cette forme et nécessite d'être exprimée par rapport à une configuration de référence connue. Le choix particulier d'une configuration de référence permet quelques simplifications. Ce choix n'affecte pas généralement les résultats du calcul.

Il s'agit de choisir comme configuration de référence une configuration adaptée au problème à traiter.

Le principe des travaux virtuels s'exprime avec des contraintes, des déformations, des volumes et des surfaces se rapportant à la configuration C_2 , qui est inconnue *a priori*. On doit donc se ramener à des configurations connues, soit C_0 en description Lagrangienne totale, soit C_1 en description Lagrangienne actualisée. Pour notre formulation, nous retiendrons la formulation Lagrangienne totale [12].

3.3.2. Formulation Lagrangienne totale

En description Lagrangienne totale, on adopte comme configuration de référence, la configuration initiale C_0 . L'utilisation de cette description nous conduit à écrire le travail virtuel de l'équation (3.1) sous la forme suivante ;

$$\delta W = \int_{0V} {}^2_0 S_{ij} \delta({}^2_0 E_{ij}) d {}^0V - \int_{0V} {}^2_0 f_i \delta u_i d {}^0V - \int_{0S} {}^2_0 t_i \delta u_i d {}^0S = 0 \quad (3.3)$$

Avec ${}^2_0 f_i$ est le vecteur des forces volumiques et ${}^2_0 t_i$ est le vecteur des forces surfaciques rapportés à la configuration C_0 .

A un instant donné, le tenseur des déformations virtuelles en chaque point du corps est :

$$\delta({}^2_0 E_{ij}) = \delta({}^1_0 E_{ij}) + \delta({}_0 \varepsilon_{ij}) = \delta({}_0 \varepsilon_{ij}) \quad (3.4)$$

Où $\delta({}^1_0 E_{ij}) = 0$

Donc le tenseur des déformations virtuelles s'exprime classiquement par la somme de deux tenseurs :

$$\delta({}_0 \varepsilon_{ij}) = \delta({}_0 e_{ij}) + \delta({}_0 \eta_{ij}) \quad (3.5)$$

Où la partie linéaire du tenseur des déformations virtuelles est donnée par :

$$\delta({}_0 e_{ij}) = \frac{1}{2} \left(\frac{\partial \delta u_i}{\partial {}^0 x_j} + \frac{\partial \delta u_j}{\partial {}^0 x_i} + \frac{\partial \delta u_k}{\partial {}^0 x_i} \frac{\partial {}^1_0 u_k}{\partial {}^0 x_j} + \frac{\partial {}^1_0 u_k}{\partial {}^0 x_i} \frac{\partial \delta u_k}{\partial {}^0 x_j} \right) \quad (3.6)$$

et la partie quadratique du tenseur des déformations virtuelles traduisant la non linéarité géométrique est :

$$\delta({}_0\eta_{ij}) = \frac{1}{2} \left(\frac{\partial \delta u_k}{\partial {}^0x_i} \frac{\partial u_k}{\partial {}^0x_j} + \frac{\partial u_k}{\partial {}^0x_i} \frac{\partial \delta u_k}{\partial {}^0x_j} \right) \quad (3.7)$$

Les tenseurs des contraintes de Piola-Kirchhoff de seconde espèce dans les configurations C_1 et C_2 sont respectivement ${}_0^1S_{ij}$ et ${}_0^2S_{ij}$. Ces deux tenseurs sont liés par la relation suivante :

$${}_0^2S_{ij} = {}_0^1S_{ij} + {}_0S_{ij} \quad (3.8)$$

Où ${}_0S_{ij}$ est le tenseur des contraintes de Kirchhoff.

Remplaçons les équations (3.4) et (3.8) dans l'équation (3.3), on obtient l'expression suivante :

$$\begin{aligned} \int_{0_V} {}_0^2S_{ij} \delta({}_0^2E_{ij}) d^0V - \delta({}_0^2R) &= \int_{0_V} ({}_0^1S_{ij} + {}_0S_{ij}) \delta({}_0\varepsilon_{ij}) d^0V - \delta({}_0^2R) = 0 \\ \int_{0_V} \{ {}_0S_{ij} \delta({}_0\varepsilon_{ij}) + {}_0^1S_{ij} [\delta({}_0e_{ij}) + \delta({}_0\eta_{ij})] \} d^0V - \delta({}_0^2R) &= 0 \quad (3.9) \\ \int_{0_V} {}_0S_{ij} \delta({}_0\varepsilon_{ij}) d^0V + \int_{0_V} {}_0^1S_{ij} \delta({}_0\eta_{ij}) d^0V + \delta({}_0^1R) - \delta({}_0^2R) &= 0 \end{aligned}$$

Où $\delta({}_0^1R)$ est l'énergie virtuelle interne du corps dans la configuration C_1 exprimé par :

$$\delta({}_0^1R) = \int_{0_V} {}_0^1S_{ij} \delta({}_0e_{ij}) d^0V \quad (3.10)$$

Le corps solide est en équilibre dans la configuration C_1 . En appliquant le principe des travaux virtuels pour la configuration C_1 , nous avons la relation suivante :

$$\int_{0_V} {}_0^1S_{ij} \delta({}_0e_{ij}) d^0V - \int_{0_V} {}_0^1f_i \delta u_i d^0V - \int_{0_S} {}_0^1t_i \delta u_i d^0S = 0 \quad (3.11)$$

Avec

$$\delta({}_0^1R) = \int_{0_V} {}_0^1f_i \delta u_i d^0V + \int_{0_S} {}_0^1t_i \delta u_i d^0S \quad (3.12)$$

Remplaçons la loi constitutive dans l'équation (3.9) on obtient :

$$\int_{0_V} {}_0C_{ijkl} \varepsilon_{kl} \delta({}_0\varepsilon_{ij}) d^0V + \int_{0_V} {}_0^1S_{ij} \delta({}_0\eta_{ij}) d^0V = \delta({}_0^2R) - \delta({}_0^1R) \quad (3.13)$$

Nous avons la simplification suivante :

$${}_0S_{ij} \approx {}_0C_{ijkl} {}_0e_{ij} \text{ et } \delta({}_0\varepsilon_{ij}) \approx \delta({}_0e_{ij}) \quad (3.14)$$

Donc l'équation (3.13) se simplifiée comme suit :

$$\int_{0V} {}_0C_{ijkl} {}_0e_{kl} \delta({}_0e_{ij}) d^0V + \int_{0V} {}^1S_{ij} \delta({}_0\eta_{ij}) d^0V = \delta({}_0^2R) - \delta({}_0^1R) \quad (3.15)$$

L'équation (3.15) est la forme faible pour le modèle élément fini basé sur la formulation Lagrangienne totale. Les composantes de contrainte ${}^1S_{ij}$ sont évaluées en utilisant la relation constitutive suivante :

$${}^1S_{ij} = {}_0C_{ijkl} {}^1E_{kl} \quad (3.16)$$

Où ${}^1E_{kl}$ est le tenseur des déformations de Green-Lagrange défini par :

$${}^1E_{ij} = \frac{1}{2} \left(\frac{\partial {}^1u_i}{\partial {}^0x_j} + \frac{\partial {}^1u_j}{\partial {}^0x_i} + \frac{\partial {}^1u_k}{\partial {}^0x_i} \frac{\partial {}^1u_k}{\partial {}^0x_j} \right) \quad (3.17)$$

3.3.3. Approximation des champs de déplacements et de déformations

Dans cette section, nous nous intéressons à la discrétisation par la méthode des éléments finis utilisant On a adopté l'hypothèse des petites déformations en grands déplacements pour les problèmes tridimensionnels non linéaires géométriques en utilisant la formulation Lagrangienne totale. Soit les notations suivantes [22] :

$$({}^0x_1, {}^0x_2, {}^0x_3) = (x, y, z) ; ({}^1u_1, {}^1u_2, {}^1u_3) = (u, v, w) ; ({}^1\bar{u}_1, {}^1\bar{u}_2, {}^1\bar{u}_3) = (\bar{u}, \bar{v}, \bar{w}) \quad (3.18)$$

On peut écrire le premier terme de l'équation (3.15) sous la forme suivante :

$$\int_{0V} {}_0C_{ijkl} {}_0e_{kl} \delta({}_0e_{ij}) d^0V = \int_{0V} \{\delta_0e\}^T [{}_0C] \{{}_0e\} d^0V \quad (3.19)$$

Avec

$$\{{}_0e\} = \begin{Bmatrix} {}_0e_{xx} \\ {}_0e_{yy} \\ {}_0e_{zz} \\ 2{}_0e_{xy} \\ 2{}_0e_{xz} \\ 2{}_0e_{yz} \end{Bmatrix} = \begin{Bmatrix} \bar{u}_{,x} + u_{,x} \bar{u}_{,x} + v_{,x} \bar{v}_{,x} + w_{,x} \bar{w}_{,x} \\ \bar{v}_{,y} + u_{,y} \bar{u}_{,y} + v_{,y} \bar{v}_{,y} + w_{,y} \bar{w}_{,y} \\ \bar{w}_{,z} + u_{,z} \bar{u}_{,z} + v_{,z} \bar{v}_{,z} + w_{,z} \bar{w}_{,z} \\ \bar{u}_{,x} + \bar{v}_{,y} + u_{,x} \bar{u}_{,y} + v_{,x} \bar{v}_{,y} + w_{,x} \bar{w}_{,y} + u_{,y} \bar{u}_{,x} + v_{,y} \bar{v}_{,x} + w_{,y} \bar{w}_{,x} \\ \bar{u}_{,x} + \bar{w}_{,z} + u_{,x} \bar{u}_{,z} + v_{,x} \bar{v}_{,z} + w_{,x} \bar{w}_{,z} + u_{,z} \bar{u}_{,x} + v_{,z} \bar{v}_{,x} + w_{,z} \bar{w}_{,x} \\ \bar{v}_{,y} + \bar{w}_{,z} + u_{,y} \bar{u}_{,z} + v_{,y} \bar{v}_{,z} + w_{,y} \bar{w}_{,z} + u_{,z} \bar{u}_{,y} + v_{,z} \bar{v}_{,y} + w_{,z} \bar{w}_{,y} \end{Bmatrix} \quad (3.20)$$

Où $[{}_0C]$ est la matrice d'élasticité pour un matériau homogène et isotrope.

On peut aussi écrire le deuxième terme de l'équation (3.15) comme suit

$$\int_{0V} {}^1S_{ij} \delta({}_0\eta_{ij}) d {}^0V = \int_{0V} \{\delta_0\eta\}^T \{{}_0^1S\} d {}^0V \quad (3.21)$$

où sous forme matricielle

$$\int_{0V} \{\delta_0\eta\}^T \{{}_0^1S\} d {}^0V = \int_{0V} \{\partial\bar{u}\}^T [{}_0^1S] \{\partial\bar{u}\} d {}^0V \quad (3.22)$$

Avec

$$[{}_0^1S] = \begin{bmatrix} {}^1\tilde{S} & \bar{0} & \bar{0} \\ \bar{0} & {}^1\tilde{S} & \bar{0} \\ \bar{0} & \bar{0} & {}^1\tilde{S} \end{bmatrix}; \quad \bar{0} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}; \quad {}^1\tilde{S} = \begin{bmatrix} {}^1S_{xx} & {}^1S_{xy} & {}^1S_{xz} \\ {}^1S_{xy} & {}^1S_{yy} & {}^1S_{yz} \\ {}^1S_{xz} & {}^1S_{yz} & {}^1S_{zz} \end{bmatrix} \quad (3.23)$$

Les champs des déplacements totaux et incrémentaux sont :

$$\{u\} = \begin{Bmatrix} u \\ v \\ w \end{Bmatrix} = \sum_{i=1}^{n_d} \begin{Bmatrix} N_i u_i \\ N_i v_i \\ N_i w_i \end{Bmatrix} \quad (3.24)$$

$$\{\bar{u}\} = \begin{Bmatrix} \bar{u} \\ \bar{v} \\ \bar{w} \end{Bmatrix} = \sum_{i=1}^{n_d} \begin{Bmatrix} N_i \bar{u}_i \\ N_i \bar{v}_i \\ N_i \bar{w}_i \end{Bmatrix} \quad (3.25)$$

Où n_d est le nombre des nœuds par élément (8 pour l'élément hexaédrique H8 et 20 pour l'élément H20).

La partie quadratique du tenseur des déformations virtuelles traduisant la non-linéarité géométrique est :

$$\{\delta_0\eta\} = \begin{Bmatrix} \delta_0\eta_{xx} \\ \delta_0\eta_{yy} \\ \delta_0\eta_{zz} \\ 2\delta_0\eta_{xy} \\ 2\delta_0\eta_{xz} \\ 2\delta_0\eta_{yz} \end{Bmatrix} = \begin{Bmatrix} \delta\bar{u}_{,x} \bar{u}_{,x} + \delta\bar{v}_{,x} \bar{v}_{,x} + \delta\bar{w}_{,x} \bar{w}_{,x} \\ \delta\bar{u}_{,y} \bar{u}_{,y} + \delta\bar{v}_{,y} \bar{v}_{,y} + \delta\bar{w}_{,y} \bar{w}_{,y} \\ \delta\bar{u}_{,z} \bar{u}_{,z} + \delta\bar{v}_{,z} \bar{v}_{,z} + \delta\bar{w}_{,z} \bar{w}_{,z} \\ \delta\bar{u}_{,x} \bar{u}_{,y} + \delta\bar{v}_{,x} \bar{v}_{,y} + \delta\bar{w}_{,x} \bar{w}_{,y} + \delta\bar{u}_{,y} \bar{u}_{,x} + \delta\bar{v}_{,y} \bar{v}_{,x} + \delta\bar{w}_{,y} \bar{w}_{,x} \\ \delta\bar{u}_{,x} \bar{u}_{,z} + \delta\bar{v}_{,x} \bar{v}_{,z} + \delta\bar{w}_{,x} \bar{w}_{,z} + \delta\bar{u}_{,z} \bar{u}_{,x} + \delta\bar{v}_{,z} \bar{v}_{,x} + \delta\bar{w}_{,z} \bar{w}_{,x} \\ \delta\bar{u}_{,y} \bar{u}_{,z} + \delta\bar{v}_{,y} \bar{v}_{,z} + \delta\bar{w}_{,y} \bar{w}_{,z} + \delta\bar{u}_{,z} \bar{u}_{,y} + \delta\bar{v}_{,z} \bar{v}_{,y} + \delta\bar{w}_{,z} \bar{w}_{,y} \end{Bmatrix} \quad (3.26)$$

Le tenseur des contraintes de Piola-Kirchhoff de seconde espèce est :

$$\{{}_0^1S\} = [{}_0C] \{{}_0^1E\} \quad (3.27)$$

Le tenseur des déformations de Green-Lagrange est :

$$\left\{ \begin{matrix} {}^1E \\ {}_0E \end{matrix} \right\} = \left\{ \begin{matrix} {}^1E_{xx} \\ {}^1E_{yy} \\ {}^1E_{zz} \\ 2 {}^1E_{xy} \\ 2 {}^1E_{xz} \\ 2 {}^1E_{yz} \end{matrix} \right\} = \left\{ \begin{matrix} u_{,x} + 0.5 \left[(u_{,x})^2 + (v_{,x})^2 + (w_{,x})^2 \right] \\ v_{,y} + 0.5 \left[(u_{,y})^2 + (v_{,y})^2 + (w_{,y})^2 \right] \\ w_{,z} + 0.5 \left[(u_{,z})^2 + (v_{,z})^2 + (w_{,z})^2 \right] \\ u_{,y} + v_{,x} + u_{,x} u_{,y} + v_{,x} v_{,y} + w_{,x} w_{,y} \\ u_{,z} + w_{,x} + u_{,x} u_{,z} + v_{,x} v_{,z} + w_{,x} w_{,z} \\ v_{,z} + w_{,y} + u_{,y} u_{,z} + v_{,y} v_{,z} + w_{,y} w_{,z} \end{matrix} \right\} \quad (3.28)$$

En injectant ces discrétisations dans les équations (3.19) et (3.21), on obtient finalement les relations suivantes :

$$\int_{0V} \{\delta_0 e\}^T [{}_0C] \{0e\} d^0V = \int_{0V} \{\delta \bar{u}_n\}^T [B_L]^T [{}_0C] [B_L] \{\bar{u}_n\} d^0V \quad (3.29)$$

$$\int_{0V} \{\partial \delta \bar{u}\}^T [{}_0S] \{\partial \bar{u}\} d^0V = \int_{0V} \{\delta \bar{u}_n\}^T [B_{NL}]^T [{}_0S] [B_{NL}] \{\bar{u}_n\} d^0V \quad (3.30)$$

on obtient le système matriciel suivant :

$$\left([{}_0^1K_L] + [{}_0^1K_{NL}] \right) \{\bar{u}_n\} = \{0^2F\} - \{0^1F\} \quad (3.31)$$

Avec

$$[{}_0^1K_L] = \int_{0V} [B_L]^T [{}_0C] [B_L] d^0V \quad (3.32)$$

$$[{}_0^1K_{NL}] = \int_{0V} [B_{NL}]^T [{}_0S] [B_{NL}] d^0V \quad (3.33)$$

$\{0^2F\}$ Représente les sollicitations extérieures indépendantes des déplacements,

$[{}_0^1K_L]$ est la matrice de rigidité linéaire,

$[{}_0^1K_{NL}]$ est la matrice de rigidité non linéaire.

Les matrices $[B_{NL}]$ et $[B_L]$ sont définies par les relations suivantes

$$[B_{NL}] = \begin{bmatrix} \tilde{B}_{NL}^u \\ \tilde{B}_{NL}^v \\ \tilde{B}_{NL}^w \end{bmatrix} \quad (3.34)$$

Et

$$[B_L] = [B_{L0}] + [B_{L1}] \quad (3.35)$$

$$\tilde{B}_{NL}^u = \begin{bmatrix} N_{i,x} & 0 & 0 & 0 \\ \cdots N_{i,y} & 0 & 0 & 0 \\ N_{i,z} & 0 & 0 & 0 \end{bmatrix} \quad \cdots i = 1, n_d \quad (3.36)$$

$$\tilde{B}_{NL}^v = \begin{bmatrix} 0 & N_{i,x} & 0 & 0 \\ \cdots 0 & N_{i,y} & 0 & 0 \\ 0 & N_{i,z} & 0 & 0 \end{bmatrix} \quad \cdots i = 1, n_d \quad (3.37)$$

$$\tilde{B}_{NL}^w = \begin{bmatrix} 0 & 0 & N_{i,x} & 0 \\ \cdots 0 & 0 & N_{i,y} & 0 \\ 0 & 0 & N_{i,z} & 0 \end{bmatrix} \quad \cdots i = 1, n_d \quad (3.38)$$

Sachant que la matrice $[B_{L0}]$ est la même matrice déformation-déplacement $[B]$ définie par la matrice $[B_{L1}]$ est défini par :

$$[B_{L1}] = [\cdots B_{L1i}^U \vdots B_{L1i}^\theta \cdots i = 1, n_d] \quad (3.39)$$

avec

$$[B_{L1i}^U] = \begin{bmatrix} l_{11}N_{i,x} & l_{21}N_{i,x} & l_{31}N_{i,x} \\ l_{12}N_{i,y} & l_{22}N_{i,y} & l_{32}N_{i,y} \\ l_{13}N_{i,z} & l_{23}N_{i,z} & l_{33}N_{i,z} \\ l_{11}N_{i,y} + l_{12}N_{i,x} & l_{21}N_{i,y} + l_{22}N_{i,x} & l_{31}N_{i,y} + l_{32}N_{i,x} \\ l_{11}N_{i,z} + l_{13}N_{i,x} & l_{21}N_{i,z} + l_{23}N_{i,x} & l_{31}N_{i,z} + l_{33}N_{i,x} \\ l_{12}N_{i,z} + l_{13}N_{i,y} & l_{22}N_{i,z} + l_{23}N_{i,y} & l_{32}N_{i,z} + l_{33}N_{i,y} \end{bmatrix} \quad (3.40)$$

où

$$l_{ij} = \sum_{k=1}^{n_d} N_{k,j} u_i^k \quad (3.41)$$

Les dérivées des fonctions de forme sont obtenues par les relations standards Suivantes :

$$\begin{aligned} \langle N_{i,x} \rangle &= j_{11} \langle N_{i,\xi} \rangle + j_{12} \langle N_{i,\eta} \rangle + j_{13} \langle N_{i,\zeta} \rangle \\ \langle N_{i,y} \rangle &= j_{21} \langle N_{i,\xi} \rangle + j_{22} \langle N_{i,\eta} \rangle + j_{23} \langle N_{i,\zeta} \rangle \\ \langle N_{i,z} \rangle &= j_{31} \langle N_{i,\xi} \rangle + j_{32} \langle N_{i,\eta} \rangle + j_{33} \langle N_{i,\zeta} \rangle \end{aligned} \quad (3.42)$$

où $l_{ij} = \sum_{k=1}^{n_d} N_{k,j} u_i^k$ sont les termes de la matrice Jacobienne inverse $[j] = [J]^{-1}$

$$[J] = \begin{bmatrix} x_{,\xi} & y_{,\xi} & z_{,\xi} \\ x_{,\eta} & y_{,\eta} & z_{,\eta} \\ x_{,\zeta} & y_{,\zeta} & z_{,\zeta} \end{bmatrix} \quad (3.43)$$

Finalement, le vecteur des forces internes s'exprime comme suit :

$$\{ {}^1_0 F \} = \int_{{}^0V} [B_L]^T \{ {}^1_0 \hat{S} \} d^0V \quad (3.44)$$

Ou

$$d^0V = \det J \, d\xi \, d\eta \, d\zeta \quad (3.45)$$

et le vecteur des contraintes de Piola-Kirchhoff de seconde espèce est :

$$\{ {}^1_0 \hat{S} \}^T = \begin{bmatrix} {}^1_0 S_{xx} & {}^1_0 S_{yy} & {}^1_0 S_{zz} & {}^1_0 S_{xy} & {}^1_0 S_{xz} & {}^1_0 S_{yz} \end{bmatrix} \quad (3.46)$$

3.4. Méthodes numériques de résolution

Les méthodes numériques de résolution consistent à discrétiser le problème en incréments et le résoudre de telle manière à ce que à l'intérieur de chaque incrément se déroule un processus itératif permettant d'approcher la solution. Pour le cas de chargements statiques, cette discrétisation correspond à une discrétisation en incréments de chargement. Une bibliographie importante sur ces méthodes est disponible dans tous les d'éléments finis traitant les problèmes non linéaires [12, 13, 14].

Ces méthodes sont adaptées pour résoudre les problèmes avec tous les non linéarités possibles, géométrique ou matérielle. Parmi ces méthodes, la plus classique est la méthode de Newton-Raphson.

La résolution directe du système (3.31) est remplacée par la recherche itérative du champ des déplacements $\{u_n\}$ qui annulera le vecteur résidu R à la fin de chaque incrément de chargement. On peut alors définir le vecteur résidu par :

$$R(\{u_n\}) = [K_L + K_{NL}] \{u_n\} - F \quad (3.47)$$

Ce vecteur résidu peut aussi s'interpréter comme la différence entre les efforts intérieurs et les efforts extérieurs imposés :

$$R(\{u_n\}) = \{F_{ext}\} - \{F_{int}\} = \{0\} \quad (3.48)$$

Notons $\{u_n\}^i$ une approximation supposée connue de la solution à l'itération (i) d'un incrément de chargement (j) donné, on peut alors calculer :

$$R(\{u_n\}^i) = \{F_{ext}\}^j - \{F_{int}\}_i^j = \sum_{k=1}^{n_d} (\{F_{int}(\sigma)\}_k)_i^j \quad (3.49)$$

La méthode consiste alors à rechercher la solution $\{u_n\}^{i+1}$ à l'itération ($i+1$) telle que :

$$R(\{u_n\}^{i+1}) = \{0\} \quad (3.50)$$

$$\{u_n\}^{i+1} = \{u_n\}^i + \{\Delta u_n\}^{i+1} \quad (3.51)$$

En développant en série de Taylor le vecteur résidu $R(\{u_n\})$ au voisinage de $\{u_n\}^i$ on obtient :

$$R(\{u_n\}^{i+1}) = R(\{u_n\}^i) + \frac{\partial R}{\partial u}(\{u_n\}^i) \cdot \{\Delta u_n\}^{i+1} + \dots = \{0\} \quad (3.52)$$

Une approximation au premier ordre conduit à l'expression suivante :

$$[K_T] \{\Delta u_n\}^{i+1} = \{F_{ext}\} - \{F_{int}\}_i \quad (3.53)$$

La solution est donc de trouver $\{\Delta u_n\}^{i+1}$ tel que :

$$\{\Delta u_n\}^{i+1} = [K_T]^{-1} \cdot R(\{u_n\}^i) \quad (3.54)$$

où $[K_T]$ désigne la matrice tangente donnée comme suit :

$$[K_T] = - \left[\frac{\partial R}{\partial u}(\{u_n\}^i) \right] \quad (3.55)$$

Partant d'une valeur initiale $\{u_n\}^0$ fixée par l'utilisateur, le processus itératif de résolution est poursuivi jusqu'à la convergence. Afin de contrôler le processus itératif, il est nécessaire d'introduire des tests de convergence. On utilise ceux portant sur les rapports de normes des déplacements, des forces résiduelles et de l'énergie au cours d'une itération. Au moins l'un des critères suivants doit être satisfait :

$$\frac{\|\{\Delta u_n\}_i\|}{\|\{u_n\}_i\|} < \varepsilon_1 \quad (3.56)$$

$$\frac{\|\{R\}_i\|}{\|\{R\}_0^j\|} < \varepsilon_2 \quad (3.57)$$

$$\frac{|\{R\}_i \langle \Delta u_n \rangle_i|}{|\{R\}_0^j \langle u_n \rangle_i|} < \varepsilon_3 \quad (3.58)$$

où $\|\bullet\|$ représente la norme. Les tolérances ε_1 , ε_2 , et ε_3 sont des données du calcul choisies par l'utilisateur (en général prises égales à 0.1%).

CHAPITRE 4

Cas tests non linéaire

4.1. Introduction

Dans la plupart, sinon la totalité, des travaux antérieurs sur la formulation des éléments finis et les procédures de solutions non linéaires, les résultats des problèmes de référence non linéaires géométriques sont présentés sous la forme de courbes de charge-déflexion. Dans ce chapitre, trois séries de problèmes de référence communément employées sont identifiées et leurs solutions de référence détaillées sont obtenues et compilées utilisant le code de calcul Abaqus. On espère que ces solutions constituent une base pratique pour comparaison ultérieure, de la reconstruction des points de données par mesure de graphique précédemment rapportés courbes charge-flèche peut être évitée. Par ailleurs, la difficulté convergente relative des problèmes sont révélés par le nombre d'incrément de charge et le nombre total d'itérations nécessaires par un régime d'incrément automatique de la charge de la réalisation des solutions de convergence sous les charges maximales.

D'examiner ou de démontrer l'exactitude des nouveaux modèles éléments finis et l'efficacité des nouvelles procédures de solutions non linéaires, des problèmes de benchmark populaires sont souvent exercés et les prédictions sont comparées à des solutions de référence. Depuis solutions analytiques de problèmes d'obus sont très limitées, la plupart des solutions de référence sont déjà rapporté des solutions numériques. Pour les problèmes de repères linéaires, ces solutions peuvent être facilement et précisément exprimées en termes numériques [15.16.17].

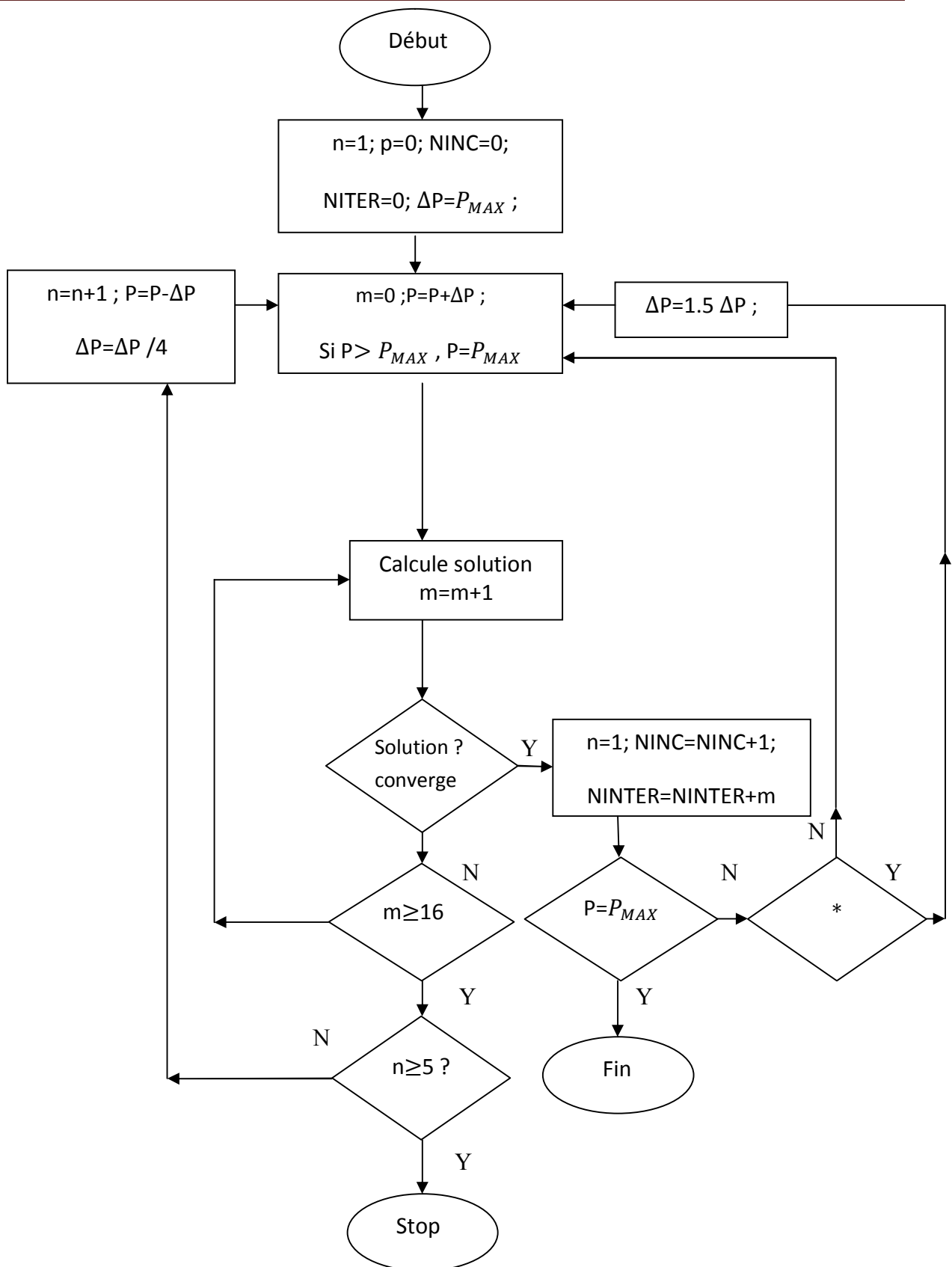
Pour mettre en évidence la difficulté de convergence relative des problèmes, le nombre

D'incrément de charge et le nombre total d'itérations requises par le schéma d'incrément automatique de la charge par défaut d'ABAQUS pour atteindre les solutions de convergence sous les charges maximales sont signalées.

4.2. Schéma d'incrément de charge

Dans la procédure de solution non-linéaire, la méthode de Newton-Raphson complète est utilisée. Les critères de convergence par défaut sont toujours utilisés et ils sont : la tolérance de force est 0,5% et la tolérance de déplacement de est 1%. Le schéma d'incrément automatique de la charge par défaut dans ABAQUS est adopté et la procédure est décrite dans la figure 4.1. Tout au long du programme, la P_{MAX} de charge maximale sera divisée automatiquement en incréments de charge NINC qui ne sont pas nécessairement uniforme. A la fin de chaque incrément de charge, on obtient une solution intermédiaire convergente. Cela permet de réduire le degré de non-linéarité d'un état de solution intermédiaire à l'autre et améliore les chances d'obtenir la solution ultime. Ce dernier est le seul sous la P_{MAX} de charge maximale. Le programme commence avec le processeur d'incrément de charge fixé à la limite de charge maximale P_{MAX} [18].

Si la solution ne converge pas dans les 16 itérations (compté de m) ou si la solution diverge, le système abandonne l'incrément et commence à nouveau avec l'incrément de charge réduit à un quart de la valeur précédente. Si la solution ne parvient toujours pas à converger, le régime réduit encore la taille de l'incrément de nouveau. Si la solution échoue après 5 tentatives (compté par n) de la réduction de l'incrément de charge, l'analyse sera arrêtée ou interrompu. D'autre part, il augmente automatiquement l'incrément de charge de 50% si les deux dernières solutions convergentes sont tous les deux obtenus dans les 5 itérations. Si le régime n'est pas échoué, ABAQUS sorties NINC et NITER. Ce dernier est le nombre total d'itérations nécessaires pour obtenir des solutions intermédiaires.



* : $m < 5$ dans les deux dernières solutions convergées ; n : conteur de réduction de charge
 m, NITER : conteurs d'itération ; NINC : conteur d'incrément de charge

Figure 4.1 : Schéma d'incrément de charge automatique [21]

4.3. Comparaison numérique en analyse non linéaire

Dans cette section, nous examinons le comportement des éléments volumiques hexaédriques à 8 nœuds C3D8 et à 20 nœuds C3D20 d'Abaqus en non linéaires géométriques et nous discutons de l'efficacité de ces éléments en comparaison avec l'élément coque à 4 nœuds S4R disponibles dans Abaqus présenté par Sze et al [18]. Un certain nombre de tests représentatifs seront analysés : Tout d'abord, nous avons étudié le comportement d'une poutre console soumise à un effort tranchant. Ce test faisant intervenir principalement de la flexion est assez sévère sur la convergence des modèles. Ensuite, le cas d'une semi coque cylindrique soumise à une force ponctuelle finalement, nous étudions aussi l'exemple de coque cylindrique sans diaphragmes soumise une force ponctuelle.

4.3.1. Poutre console soumise à un effort tranchant

La figure 4.2 montre une poutre encastree à l'une de ses extrémités et soumise à une charge linéique P uniformément répartie sur l'extrémité opposée. Ce test est présenté par Sze et al [18] afin de vérifier la bonne performance de l'élément de coque S4R du code de calcul éléments finis Abaqus avec un maillage de 16×1 éléments. Cette solution est considérée comme une solution de référence.

En regardant le comportement du modèle face à un chargement induisant de grands déplacements, on va solliciter la poutre par une charge P_{MAX} par incréments de 20 %.

$$P_0 = EI/L^2 = 1 \text{ et } P_{MAX} = 4 P_0$$

Le calcul est effectué avec un maillage de $(100 \times 10 \times 1)$ éléments hexaédriques. Les résultats obtenus sont reportés dans la figure 4.3, Cette figure représente les courbes de la force normalisée en fonction du déplacement au point d'extrémité libre dans les deux directions Ox et Oz . Nous comparons les résultats obtenus par les éléments volumiques linéaires classiques C3D8 et C3D20 avec la solution de référence. Les figures 4.4 et 4.5 montre le maillage initial et déformé de la poutre console soumise à une charge linéique maximale P_{max} .

Données
 $E = 1.2 \times 10^6$
 $\nu = 0$
 $L = 10$
 $b = 1$
 $h = 0.1$
 $P_0 = EI/L^3$
Chargement
 $P_{max} = 4P_0 = 1$

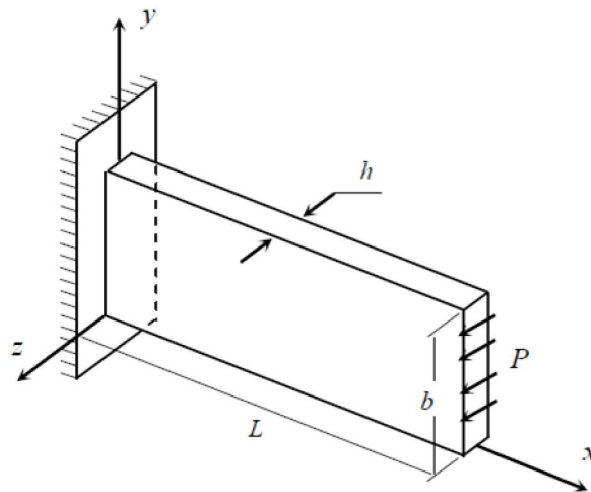


Figure 4.2 : Poutre console en non linéaire géométrique.

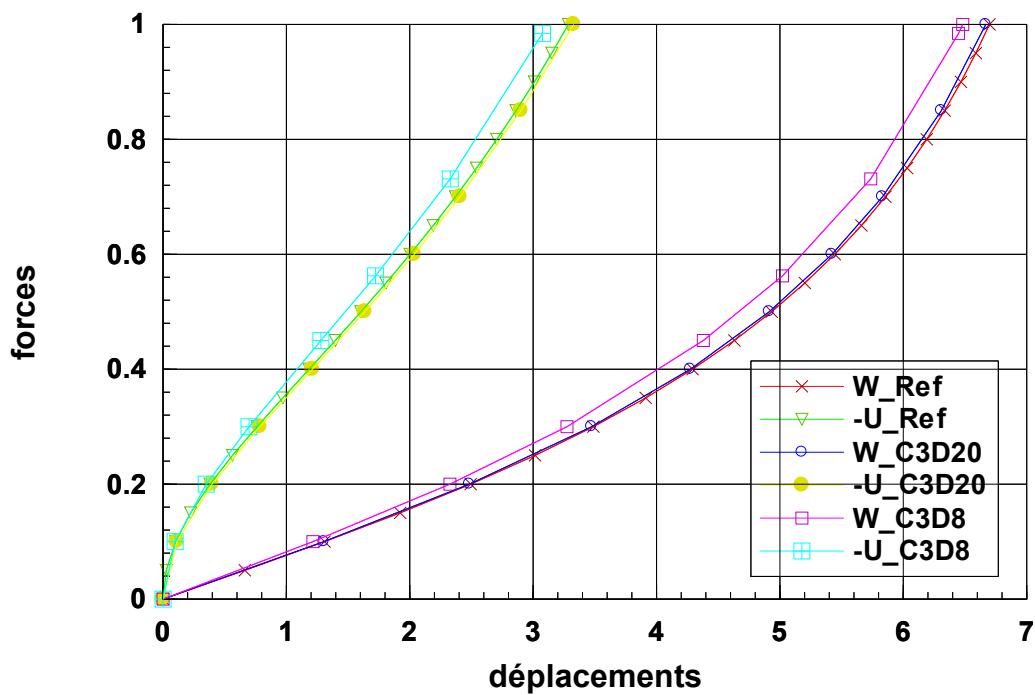


Figure 4.3 : Résultats de la poutre consol en non linéaire géométrique avec les éléments C3D8 et C3D20

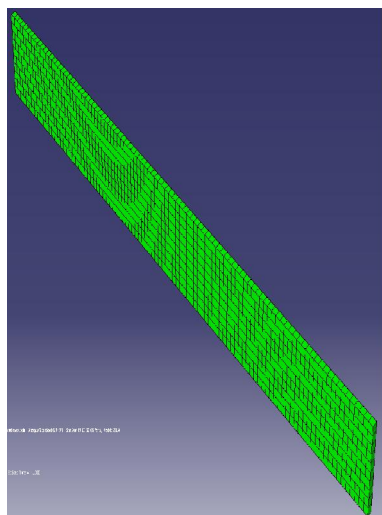


Figure 4.4 : Configuration initial ($P = 0$) de la poutre

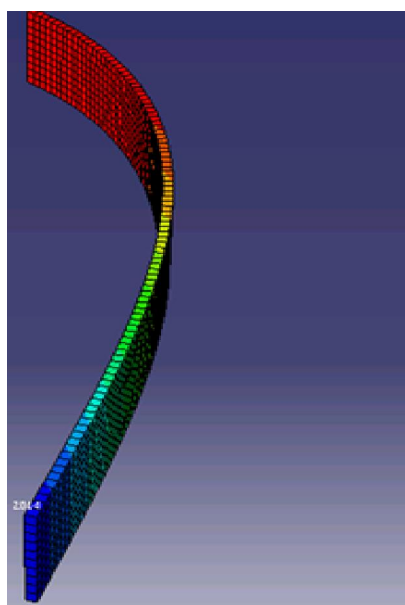


Figure 4.5 : Déformation de la poutre consol ($P = P_{\max}$)

4.3.2. Coque semi cylindrique

La figure 4.6 montre une coque encastrée à coté et soumise à une force P posée sur le point A de la coque. Cette structure est modélisée avec un maillage de $(40 \times 40 \times 1)$ éléments hexaédriques. Les résultats obtenus avec les éléments volumiques C3D8 et C3D20 et l'élément coque de référence S4R sont tous regroupés dans la figure 4.7. Ces courbes force-déplacement pour les deux éléments volumiques C3D8 et C3D20 montrent un bon accord avec la solution de référence obtenue avec l'élément de coque S4R. La figure 4.9 illustre le maillage déformé de la coque cylindrique à l'incrément $P_{max} = 2000$ obtenu avec l'élément hexaédrique à 20 nœuds C3D20. Ce dernier est comparé avec le maillage déformé obtenu avec l'élément coque S4R présenté par Sze et al [18] dans la figure 4.8.

Données

$$E = 2.085 \times 10^7$$

$$L = 3.048 \text{ m}$$

$$h = 0.03 \text{ m}$$

$$R = 1.016 \text{ m}$$

$$\nu = 0.3$$

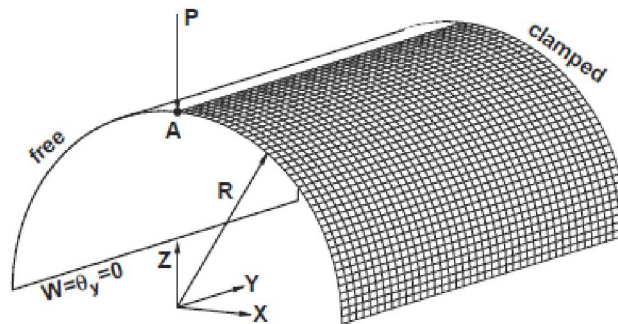


Figure 4.6. Coque semi cylindrique

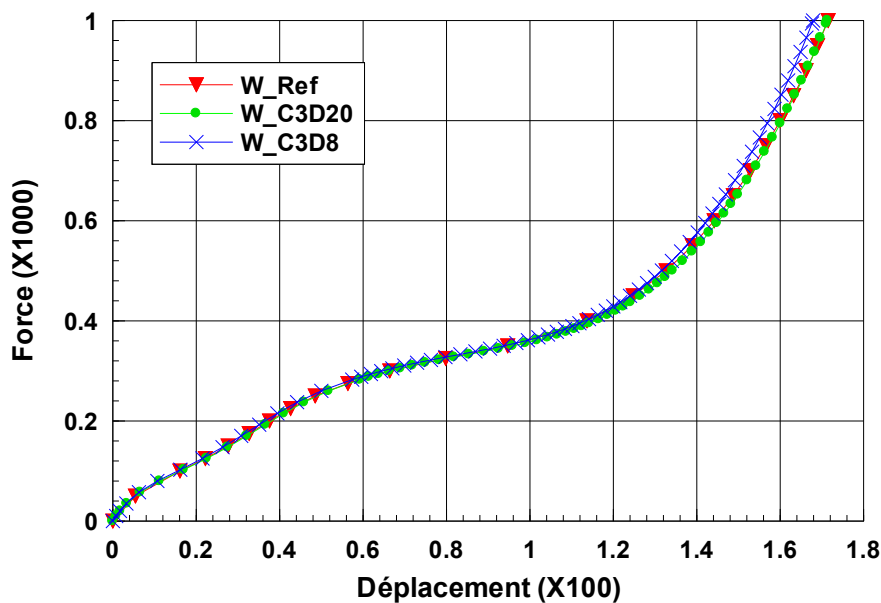


Figure 4.7 : Courbes force-déplacement de la coque pour les deux éléments volumiques C3D20 et C3D8

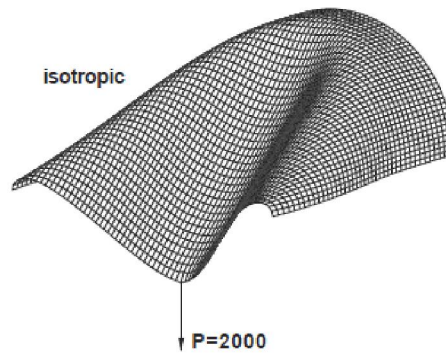


Figure 4.8 : Déformation de la structure [18]

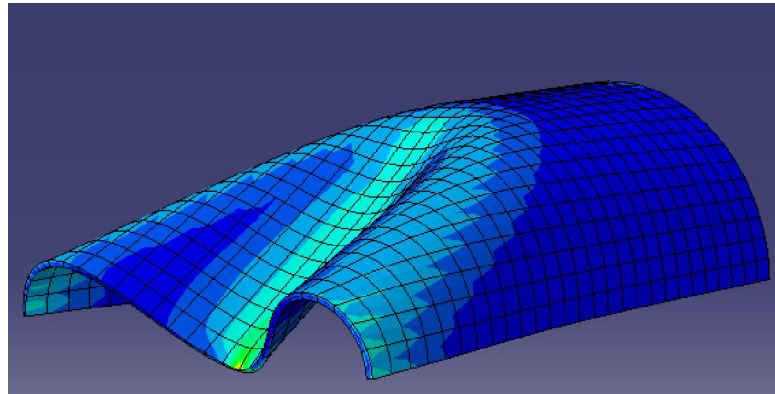


Figure 4.9 : Déformation de la structure avec élément volumique C3D20

4.3.3. Coque cylindrique sans diaphragmes soumise à une force ponctuelle

La figure 4.10 montre une coque cylindrique sans diaphragmes soumise à une charge concentrée. Ce cas test a été étudié par Sze et al [18, 19] et Ooi et al [20]. Le problème est doublement symétrique, nous avons donc étudiés seulement un huitième de la structure. La coque est libre sur le côté CD et soumise à un chargement concentré P au point A suivant la direction verticale Oz figure 4.10. La force P est appliquée par incréments successifs pour aller de 0 à P_{max} . Nous avons utilisé dans ce test un maillage de $(15 \times 22 \times 1)$ éléments volumique C3D20. Les courbes charge/déplacement du point A suivant Oz sont représentées dans la figure 4.11 La comparaison entre les résultats obtenus avec l'élément C3D20 et ceux donnés par l'élément S4R montre qu'il y a un très bon accord. Enfin, et de même que pour le premier et le deuxième exemple, nous avons présenté le maillage initial et déformé du quart de la coque sphérique pour différents incréments de chargement dans les figures 4.12 et 4.13.

Données

$$E = 10.5 \cdot 10^6$$

$$\nu = 0.3125$$

$$R = 4.953$$

$$L = 10.35$$

$$h = 0.094$$

Chargement

$$P_{max} = 40.000$$

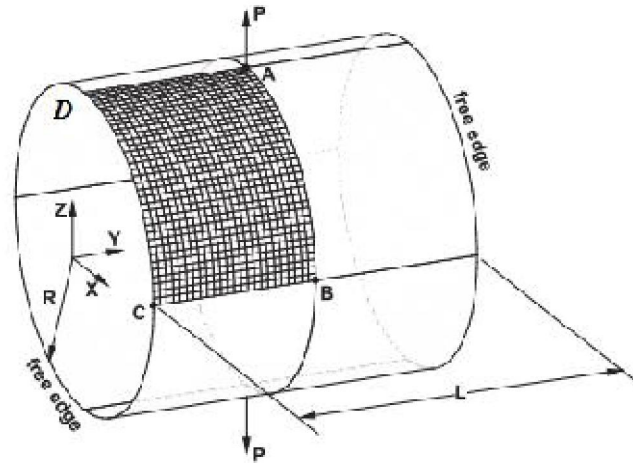


Figure 4.10 : Coque cylindrique soumise à une force ponctuelle. Géométrie, chargement et Conditions aux limites [18]

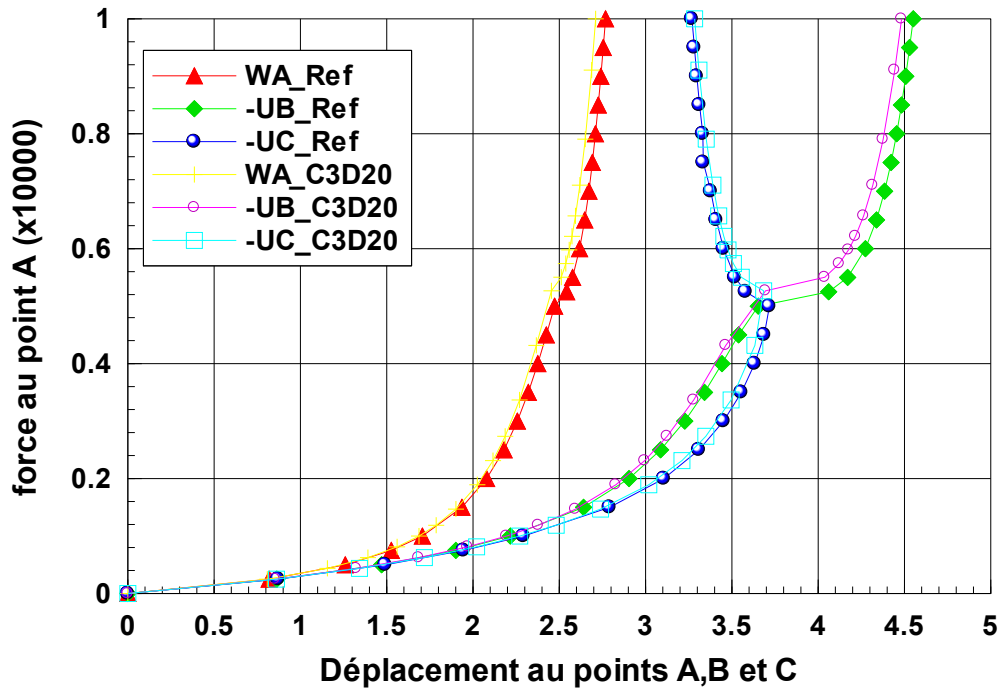


Figure 4.11 : Déflexion W_A de la coque cylindrique soumise à une charge ponctuelle.
Résultats de l'élément C3D20

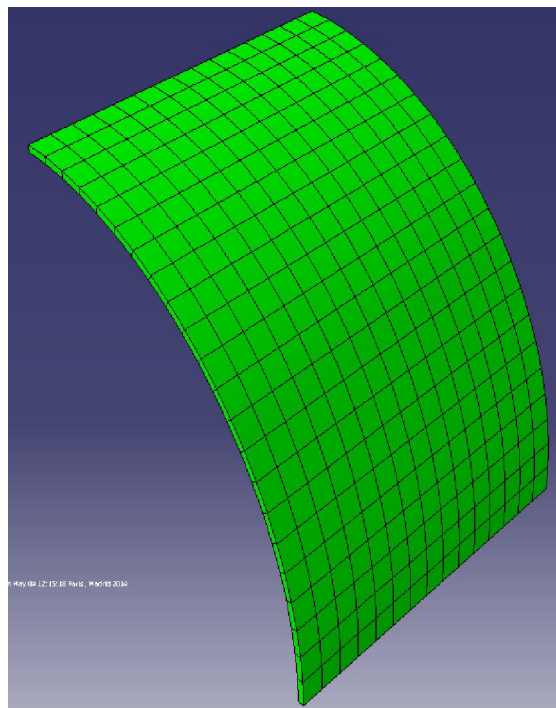


Figure 4.12 : Coque cylindrique en position initial

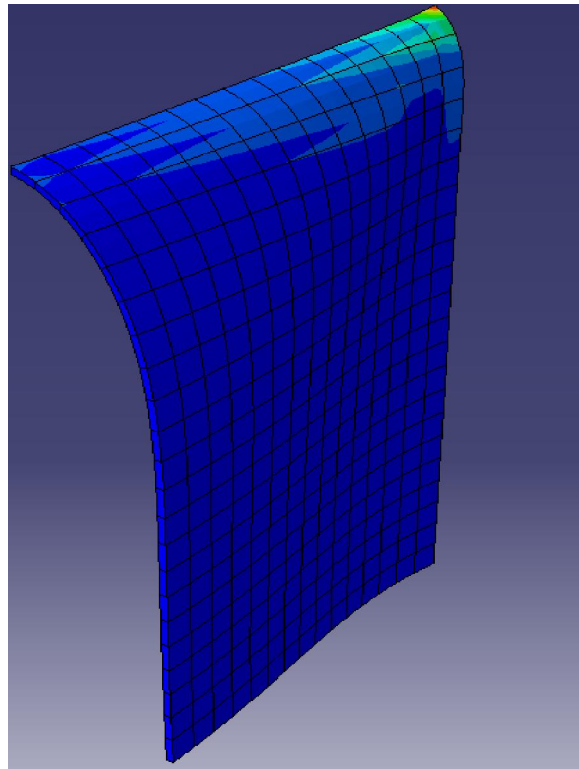


Figure 4.13 : Maillage déformé du cylindre ouvert sous la charge maximale

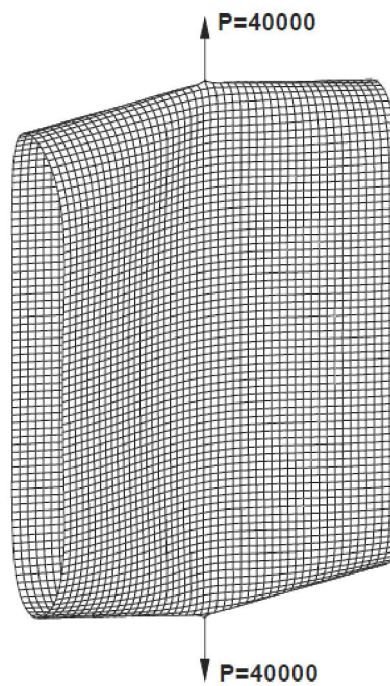


Figure 4.14 : Maillage déformé du cylindre ouvert sous la charge maximale maillage (24×36)

[18]

Conclusion

Conclusion générale

L'objectif de ce travail a été de simuler des cas tests, généralement tridimensionnels, non linéaires géométrique en grands déplacement et petites déformations rencontrés lors de la modélisation des solides et des structures par des éléments finis volumiques adoptés aux problèmes. Les différents modèles utilisés sont des éléments finis volumiques classiques hexaédriques à 8 nœuds et à 20 nœuds de code de calcul Abaqus.

Dans cette étude sur les cas tests non linéaire géométrique en 3D les résultats obtenus avec les éléments volumiques C3D8 et C3D20 d'Abaqus sont confondus avec résultats de référence obtenus avec l'élément fini coque à 4 nœuds S4R avec une intégration réduite. D'après ces résultats on peut dire que nous avons obtenus des solutions de référence pour les problèmes non linéaires géométriques tridimensionnels. Ces solutions de référence peut être utilisé pour valider spécialement les nouveaux éléments finis volumiques.

Dans cette étude on a remarqué que dans ABAQUS et même dans tous les codes de calcul par éléments finis, quand le maillage est raffiné les résultats sont plus précis. Enfin, tous les problèmes étudiés sont strictement non linéaire géométriquement et la non-linéarité matérielle n'est pas considérée.

Références Bibliographiques

Références bibliographique

- [1] **Hervé Oudin**. Méthode des éléments finis ; université de Nantes .France. cel-00341772, version 1 – 26 .Nov 2008.
- [2] **G.Dhatt and G. Touzot**. Une présentation de la méthode des éléments finis. Les presses ;Université Laval, Québec .1981. 18, 27.
- [3] **Batoz J.L. and Dhatt G**. Modélisation des structures par éléments finis. Hermès ; 1990
- [4] **DENIS ABRY**. Présentation général d’Abaqus. Formation des masters; école central de paris. 2008.
- [5] **PATRICKH ROZICHI**. Formation ABAQUS ; Université de Nantes . France 2007.
- [6] **MEFTEH KAMEL**. Cours : Introduction général au logiciel Abaqus université de Biskra. 2013.
- [7] **MEFTEH KAMEL**. Présentation Abaqus. Formation des masters ; université de Biskra .2014.
- [8] **ABAQUS**. Formation sur le code de calcul ABAQUS – module Explicit, élève ingénieur ; université de Nantes. 2007.
- [9] **STEPHANE BOUCHARÉ**. Descriptif ABAQUS ; École d'ingénieurs en génie industrie. Bretagne. 2012.
- [10] **MICKAEL ABBAS**. Principe et fonctionnement ABAQUS .université de technologie de Compiègne .2011.
- [11] **Aouameur A**. Analyse non-linéaire matérielle et géométrique des structures coques en béton armé sous chargements statiques et dynamiques: Thèse de doctorat ; Ecole Nationale des Ponts et Chaussées. France.1998.
- [12] **Reddy JN**. An introduction to nonlinear finite element analysis ; Oxford University Press, USA. 2004.

- [13] **Chrisfield M.** Non-linear Finite Element Analysis of Solids and Structures, Volume 1: Essentials; New York: John Wiley & Sons.1991.
- [14] **Bathe KJ.** Finite element procedures. Prentice hall Englewood Cliffs, NJ. 1996.
- [15] Popular benchmark problems for geometric non linear analysis of shells. Departments of Mechanical Engineering; the University of Hong Kong, China.2003.
- [16] **N.K. Prinja, R.A. Clegg.** Assembly Benchmark Tests for 3-D Beams and Shells Exhibiting Geometric Non-Linear Behaviour; NAFEMS, Glasgow, UK.1993.
- [17] **ABAQUS,** ABAQUS Theory and User's Manuals. Version 5.8.Hibbitt, Karlsson & Sorensen, Inc,Pawtucket. Rhode Island, USA.1998.
- [18] **Sze K, Liu X, Lo S.** Popular benchmark problems for geometric nonlinear analysis of shells. Finite elements in analysis and design. 2004;40:1551-69.
- [19] **Sze KY, Chan WK, Pian THH.** An eight-node hybrid-stress solid-shell element for geometric non-linear analysis of elastic shells. International Journal for Numerical Methods in Engineering. 2002;55:853-78.
- [20] **Ooi ET, Rajendran S, Yeo JH.** Extension of unsymmetric finite elements US-QUAD8 and US-HEXA20 for geometric nonlinear analyses. Engineering Computations. 2007;24:407.
- [21] **K.Y. Sze et al.** Popular benchmark problems for geometric nonlinear analysis of shells. Finite Elements in Analysis and Design 40 (2004) 1551–1569.
- [22] **MEFTAH KAMEL.** Modélisation numérique des solides par éléments finis volumiques basés sur le concept SFR (Space Fiber Rotation). PhD thesis, Université de Biskra, Algérie (2013) .