

Université catholique de Louvain
Faculté des sciences appliquées
Unité de génie civil et environnemental



**Modélisation numérique
de l'interaction fluide-structure
pour des membranes hyperélastiques
en grandes déformations**

Promoteur : **J-F. Remacle**
Lecteurs : **N. Chevaugéon
E. Marchandise
B. Spinewine
V. Legat**

Travail de fin d'études présenté par

**Gaëtan Compère
Christophe le Polain de Waroux**

en vue de l'obtention du grade
d'Ingénieur Civil des Constructions

Louvain-la-Neuve
Année académique 2004-2005

Remerciements

Nous tenons tout d'abord à remercier vivement Jean-François Remacle, pour ses conseils avisés, sa disponibilité, mais aussi pour son dynamisme et son enthousiasme, qui ont été pour nous sources d'intérêt et de motivation. La passion qu'il nous a transmise au cours des dernières années pour le calcul numérique nous a permis de mener ce travail avec l'ardeur et la motivation qui font du mémoire une expérience inoubliable.

Notre gratitude va tout particulièrement à Nicolas, pour son engagement continu, son aide précieuse tout au long de l'année, et l'intérêt indéfectible qu'il a porté à notre travail. La patience dont il a fait preuve, son soutien et sa bonne humeur nous ont aidés à maintenir le cap dans les remous, parfois chahutés, du mémoire.

Nous remercions très chaleureusement Emilie pour sa disponibilité, ses conseils pertinents, et l'aide indispensable qu'elle nous a apportée dans la compréhension de la mécanique des fluides. Ses encouragements et le regard positif qu'elle a porté sur notre travail nous ont poussés à avancer toujours plus loin.

Nos remerciements vont également à Philippe Geuzaine pour sa collaboration efficace, et l'aide immense qu'il nous a apportée en nous permettant d'utiliser un programme de calcul de fluides qu'il a patiemment développé.

Nous sommes reconnaissants envers Vincent Legat et Benoît Spinewine pour avoir accepté de consacrer du temps à la lecture de notre mémoire.

Nous souhaitons remercier nos parents, ainsi que nos frères et sœurs, pour leur soutien et pour les conseils judicieux qu'ils nous ont apportés.

Nous remercions également les équipes du kot DIH et du kot Kontre la Soif pour leurs encouragements quotidiens.

Nos derniers remerciements vont à l'ensemble des GC23 pour leur bonne humeur de tous les jours et pour la formidable ambiance qu'ils font régner dans le groupe.

Table des matières

Introduction	1
I Elaboration d'un code de calcul de membranes	6
1 Théorie des membranes en grandes déformations	7
1 Cinématique	8
1.1 Configurations eulériennes et lagrangiennes	8
1.2 Principaux tenseurs de déformation	9
1.3 Membranes axisymétriques et membranes 3D	10
1.4 Bases covariantes et contravariantes	11
1.5 Tenseurs des déformations en bases locales	13
2 Petites et grandes déformations	15
2.1 Linéarisation des déformations	15
2.2 Exemple du 1D	16
3 Comportement du matériau	17
3.1 Principes et hypothèses	17
3.2 Elongations principales et invariants du tenseur des dilatations	17
3.3 Energie de déformation	18
3.4 Relation contraintes-déformations	19
3.5 Réduction à un problème plan	21
2 Méthodes de discrétisation	24
1 Méthode des éléments finis	24

1.1	Démarche des éléments finis	24
1.2	Eléments parents	26
1.3	Eléments utilisés	27
1.4	Discrétisation des grandeurs principales	30
2	Calcul des forces internes et externes	32
2.1	Principe des Travaux Virtuels	32
2.2	Forces internes	33
2.3	Forces externes	34
2.4	Matrice de masse	34
2.5	Intégration numérique	35
3	Schémas dynamiques	37
3.1	Equation générale	37
3.2	Schémas explicites et implicites	37
3.3	Algorithme implicite : équations de Newmark	41
3.4	Schéma explicite	44
3.5	Amortissement	45
3.6	Conditions aux limites	46
3	Implémentation	47
1	Description du logiciel <i>Gmsh</i>	47
1.1	Les fonctions	47
1.2	Le maillage produit	48
2	Description du programme	49
2.1	Schéma général	49
2.2	Entrées et sorties du programme	51
2.3	Organisation du maillage	52
2.4	Mixité triangles-quadrangles	54
2.5	Imposition des conditions aux frontières	54
2.6	Imposition de conditions initiales	55
4	Vérifications et applications du code de calcul	57

1	Membrane plane	57
1.1	Membrane au repos	58
1.2	Membrane en mouvement	60
2	Cylindre infini	65
3	Membrane précontrainte	67
4	Sphère	69
5	Déformations extrêmes	71
6	Applications diverses	73
II	Couplage avec un programme de calcul de fluides	76
5	Description du programme fluide	77
1	Modèle mathématique	77
1.1	Equations fondamentales	77
1.2	Formulation ALE	79
1.3	Turbulence	80
2	Méthodes de discrétisation	81
2.1	Technique de discrétisation spatiale	82
2.2	Techniques de discrétisation des équations	82
6	Description du couplage	84
1	Fonctionnement général	84
2	Maillages 3D	86
2.1	Concordance 2D et 3D	86
2.2	Adaptation de la structure <i>mesh</i>	87
3	Communication fluide-structure	88
3.1	Transmission des forces	88
3.2	Transmission des déplacements	89
4	Prise en compte de forces extérieures quelconques	89
7	Applications	92

1	Ecoulement laminaire non-visqueux en conduite	92
2	Ecoulement de Poiseuille entre deux plans	98
3	Application médicale	101
4	Descente d'un parachute	105
5	Gonflement d'un airbag	109
Conclusion		111
Bibliographie		113

Introduction

La nécessité de pouvoir modéliser le comportement d'une membrane en présence d'un fluide, ou inversement d'un fluide au contact d'une membrane est présente dans de nombreux domaines.

Le matériau vivant présent naturellement dans le corps humain subit des déformations pouvant aller jusqu'à 40 % et a de plus un comportement non linéaire. L'élaboration d'une simulation numérique pour ce type de matériau est une étape indispensable pour permettre de mieux comprendre le comportement de ces structures que sont le cœur, les vaisseaux, les muscles, ou à une autre échelle les cellules. Examinons par exemple la problématique d'une opération cardio-vasculaire courante : le pontage. Il s'agit d'une opération lourde et risquée visant à redessiner le réseau artériel d'un patient dont le système cardio-vasculaire présente de graves défaillances. Actuellement, les chirurgiens disposent de très peu d'outils pour optimiser la nouvelle distribution des écoulements sanguins. Ils procèdent essentiellement par expérience. La mise à disposition d'un outil numérique capable de modéliser les écoulements sanguins dans les

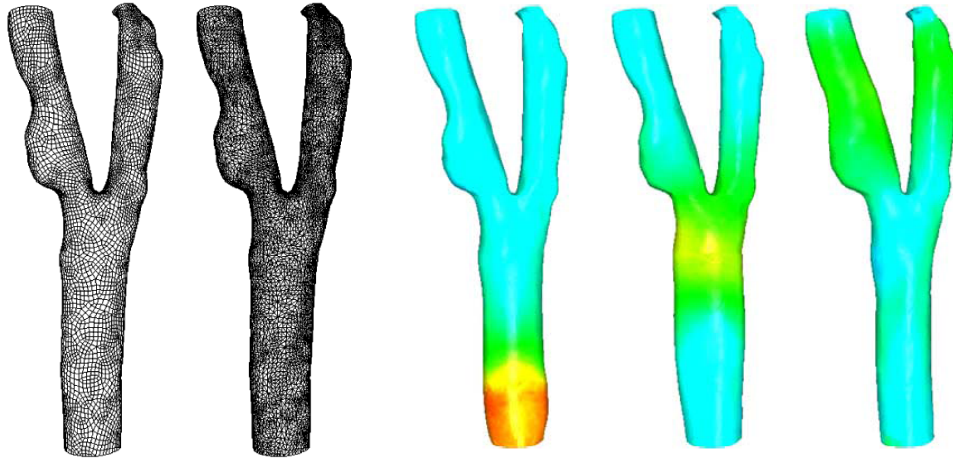


FIG. 1 – Modélisation d'une carotide via un calcul numérique d'interaction fluide-structure par Gerbeau et Vidrascu [1].

artères permettrait d'améliorer le traitement en mettant en évidence, avant l'opération,

les lacunes d'un réseau proposé, et en comparant les caractéristiques de l'écoulement et des tensions artérielles de plusieurs solutions.

Cet outil peut également s'avérer très utile, voire indispensable, dans de nombreux domaines de l'ingénieur. En aéronautique, on cherche à modéliser le comportement d'un parachute (figure 2) ou d'une aile de parapente (figure 3) ; en biomécanique, l'étude de petites pompes aspirant un fluide par le mouvement d'une membrane requiert de tels programmes ; l'automobile y trouve une utilité directe pour la conception de systèmes d'airbags et la conception de pneus ; la modélisation d'une voile de bateau, d'une toile de chapiteau soumise au vent ou de structures gonflables en sont des exemples.

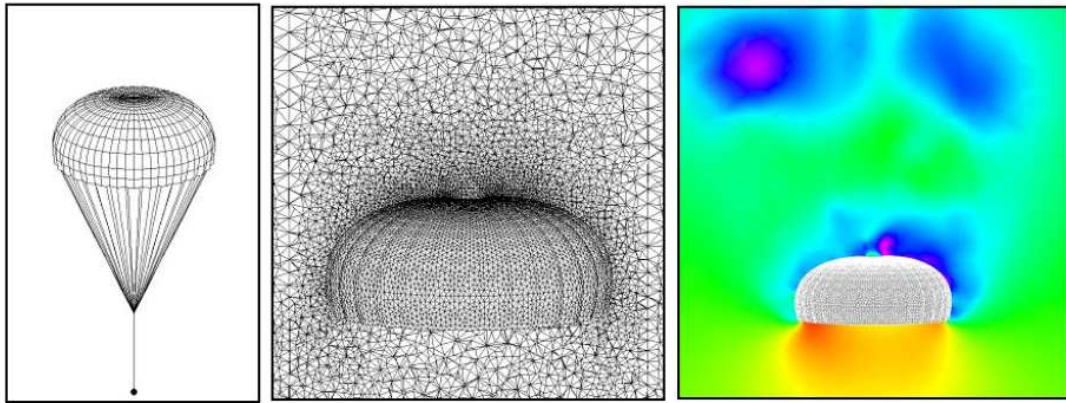


FIG. 2 – Modélisation du comportement d'un parachute via un calcul numérique en interaction fluide-structure par Stein et Tezduyar [2].

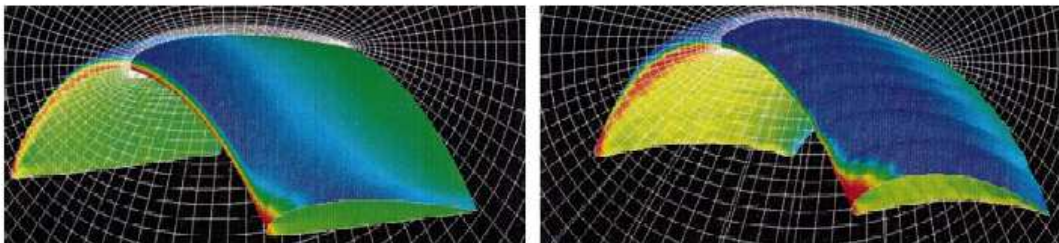


FIG. 3 – Modèle numérique d'un parapente par Kalro et Tezduyar [3].

La réalisation d'un programme de modélisation des membranes est en elle-même un ouvrage intéressant dans de nombreux domaines. Le Génie civil peut y faire appel pour modéliser le comportement d'une toile soumise à des charges (figure 4), pour étudier un joint souple, ... Des applications de la mécanique, comme le thermosoufflage, de la biomécanique, de la médecine et de bien d'autres domaines sont nombreuses. Des applications très spécifiques naissent régulièrement. Par exemple, une technique a récemment été mise au point pour combattre la presbytie, provoquée par le relâchement des muscles chargés de focaliser la vision. L'ajout de petits éléments rigides dans

l'œil a été étudié en utilisant un programme de calcul de membranes pour modéliser la cornée [4].

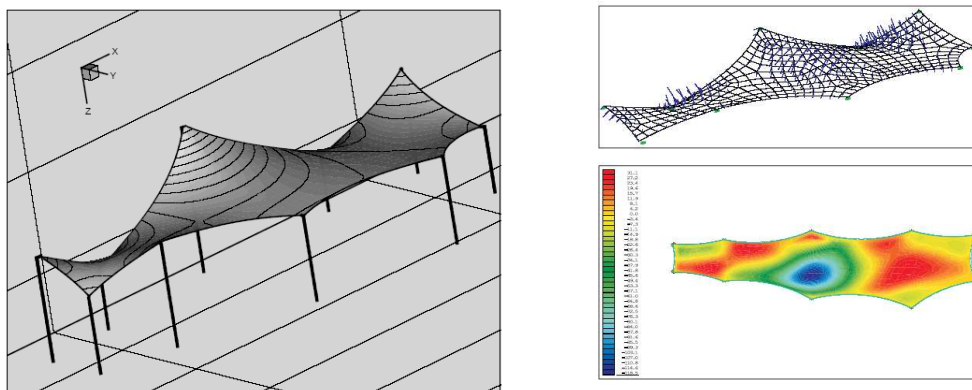


FIG. 4 – Modélisation d'une toile soumise au vent, par Markus Glück [5].

Pourquoi un modèle numérique ?

L'utilité d'une étude numérique de ces différents projets est de plus en plus claire. L'étude analytique du comportement d'une membrane, indépendamment de toute influence d'un fluide, devient extrêmement difficile dès lors qu'il s'agit d'une membrane quelque peu complexe. Quant à l'interaction fluide-membrane, la complexité d'un traitement analytique rend impossible toute étude purement théorique d'un projet précis.

Les études expérimentales sont utiles pour raffiner un objet ou confirmer ses propriétés. Elles sont toutefois coûteuses et parfois lourdes à mettre en œuvre, surtout lorsqu'il s'agit de représenter un mécanisme d'interactions entre un fluide et une membrane (contrôle des paramètres de pression, de vitesse, de débit ; écoulement tridimensionnel, problèmes d'échelle, maîtrise des conditions aux limites, ... pour le fluide ; contrôle des contraintes, des déformations ; maîtrise du matériau, des conditions aux frontières, ... pour la membrane).

La détermination préalable du comportement de l'objet est donc très utile, voire indispensable, si l'on veut éviter de dépenser beaucoup d'argent dans de longs essais. Les méthodes numériques sont un moyen peu coûteux et flexible d'établir des modèles d'interactions. Dans beaucoup de cas, il convient néanmoins de valider les modèles numériques par des modèles physiques.

Notons que ces vingt dernières années, le développement des modèles numériques s'est considérablement accéléré. Par ailleurs, l'efficacité croissante des ordinateurs permet d'exécuter des modélisations de plus en plus complexes.

Etat des lieux

Nous ne cherchons pas ici à faire une étude bibliographique à proprement parler, mais plutôt de donner quelques points de repère au lecteur à propos de la modélisation des membranes et de l'interaction fluide-membrane.

L'expression de l'énergie de déformation aujourd'hui couramment utilisée pour les membranes en grandes déformations supposées incompressibles a été écrite par Mooney (1940), puis simplifiée par Rivlin (1948). Depuis, une expression a vu le jour pour des matériaux compressibles (Simo et Pister, 1984).

Les bases du calcul des membranes en grandes déformations ont été posées par Green et Adkins avec leurs travaux sur l'élasticité non-linéaire (1960). Depuis lors de nombreux écrits ont été publiés sur le sujet. La plupart de ceux-ci traitent de l'équilibre de membranes cylindriques ou sphériques soumises à une pression uniforme (Corneliussen et Shield, 1961 ; Alexander, 1971 ; Haughton et Ogden, 1978 ; Ratner, 1983 ; Li et Steigmann, 1993 ; Haseganu et Steigmann, 1994 ; Chen, 1995). Notons que le nombre d'expériences réalisées dans ce domaine est relativement faible par rapport aux travaux théoriques et numériques.

L'analyse de grandes déformations de membranes contenant un fluide au repos a fait l'objet de quelques rares études depuis les travaux de Green et Adkins. Les publications les plus importantes sont celles de Yu et Valanis (1970), Boyer et Gutkowski (1970) et, plus récemment, Haughton (1996) et Pamplona (2000).

L'importance de l'étude du comportement statique et dynamique des membranes en biologie et en bio-ingénierie a été mise en évidence par de nombreux travaux tels que ceux de Evans et Skalak (1981), Secomb et Gross (1983), Pamplona et Calladine (1983).

La modélisation de l'interaction d'un fluide en mouvement avec des structures mobiles, mais moins déformables que les membranes a quelques années de moins. Parmi les travaux réalisés, certains sont particulièrement intéressants car ils marquent la transition vers l'étude de l'interaction fluide-membrane en tentant de répondre à des problèmes tels que la modélisation d'écoulements sanguins. Citons, parmi d'autres, les travaux de Pedley et Ralph sur les écoulements visqueux à travers un canal comportant un obstacle mobile (1985), ceux d'Ogawa et Ishiguro, qui ont développé un modèle bidimensionnel afin d'étudier les flux sanguins dans un ventricule du cœur (1987), et Rogers et al., qui ont résolu les équations de Navier-Stokes pour étudier un écoulement à travers un cœur artificiel (1991). Rosenfeld et Kwak (1989) ont appliqué une transformation des coordonnées en fonction du temps afin de résoudre les problèmes liés aux frontières mobiles, traçant ainsi le chemin de la formulation moderne *ALE* des équations de Navier-Stokes.

L'étude numérique de l'interaction fluide-membrane en régime dynamique est très récente. Peu de travaux ont été réalisés à ce jour. Epinglons toutefois les travaux de Tezduyar, Behr et Liou, qui ont développé un modèle numérique de dynamique des fluides impliquant des frontières mobiles (1992), et qui par la suite ont étudié des

modèles de parachutes par couplage fluide-structure (1999).

Objectifs du mémoire

L'objectif de ce mémoire est double :

- d'une part l'écriture, de A à Z, d'un programme de calcul des déformations dans le temps d'une membrane 3D soumise à un jeu de forces, ainsi que sa vérification à l'aide de problèmes solutionnés par la théorie. Nous nous limiterons pour ce faire à des matériaux incompressibles, homogènes et isotropes.
- d'autre part, le couplage de ce code avec un code fluide existant et l'observation de résultats d'interactions fluide-membrane.

Pour la réalisation du code, nous commencerons par étudier la théorie des grandes déformations pour les membranes. A cette occasion, nous rédigerons une démarche complète permettant d'obtenir les tenseurs, les équations et les outils nécessaires à la création du programme. Nous établirons également le lien existant avec la théorie des petites déformations.

Nous nous appuierons notamment sur la thèse de doctorat de Nicolas Chevaugnon et ses explications. Les recherches bibliographiques nécessaires seront menées afin de compléter la théorie et de la comprendre avec suffisamment de profondeur que pour se sentir à l'aise lors de son implémentation. Différents modèles de matériaux hyperélastiques seront évoqués.

Par la suite, nous utiliserons différents outils du calcul numérique (discrétisations spatiales et temporelles) afin de réaliser le code de calcul. Nous fournirons les explications nécessaires à la compréhension des démarches utilisées en tentant de leur donner un aspect didactique.

Le code sera implémenté en utilisant le bagage théorique acquis et sera vérifié via des cas solutionnés par la théorie.

Enfin, nous parcourerons la théorie de base d'un programme de calcul des fluides, et nous le couplerons avec notre code. Pour ce faire, nous acquérirons les notions de la mécanique des fluides nécessaires, et nous tenterons de traduire cet apprentissage en un exposé clair.

Nous terminerons ce travail par des applications pratiques du couplage fluide-membrane. Notons que le temps consacré à l'application du programme de couplage sera plus important que celui réservé à l'apprentissage des notions nécessaires, car la connaissance la plus poussée et la plus durable du sujet sera acquise au fur et à mesure des essais et erreurs de sa mise en pratique.

Première partie

Elaboration d'un code de calcul de
membranes

Chapitre 1

Théorie des membranes en grandes déformations

Dans ce chapitre, nous décrivons les différents outils de la mécanique des milieux continus nécessaires à l'établissement d'un modèle cinématique des membranes en grandes déformations. Nous y abordons les règles qui régissent le comportement des matériaux hyperélastiques, et notamment l'expression expérimentale de Mooney-Rivlin caractérisant l'énergie de déformation de tels matériaux. En fin de chapitre, nous établirons une relation entre les contraintes et les déformations.

Qu'est-ce qu'une membrane ?

Une membrane est une structure qui s'étend essentiellement dans deux directions de l'espace et dont l'épaisseur est faible. Du point de vue mécanique, elle présente une résistance à la traction dans son plan mais ne peut reprendre aucun effort de compression significatif. Le comportement classique d'une membrane en compression dans son plan est le froissement. Sa faible épaisseur et surtout l'absence de contraintes de compression font qu'une membrane ne présente pas de résistance à la flexion. Des feuilles de papier, des bandes magnétiques, des parois artérielles et des ballons en caoutchouc en sont des exemples.

Qu'est-ce qu'un matériau hyperélastique ?

Un objet possède un comportement élastique si, libéré après avoir subi une déformation suite à l'application d'une force, il revient de lui-même dans son état initial. Un matériau hyperélastique est un matériau capable de subir une grande déformation tout en conservant un comportement élastique. Le caoutchouc est l'exemple type du matériau hyperélastique. Les équations constitutives sont caractérisées par un comportement non-linéaire, et le matériau est souvent incompressible.

1 Cinématique

1.1 Configurations eulériennes et lagrangiennes

Notions générales

En mécanique des milieux continus, on distingue deux modèles de représentation d'une grandeur \mathbf{G} , qu'elle soit scalaire, vectorielle ou tensorielle :

- la représentation **eulérienne**, dans laquelle on considère à chaque instant la grandeur en chacun des points. Toutes les données de l'espace sont considérées au temps t , indépendamment des particules se trouvant à l'endroit étudié :

$$\mathbf{G} = \mathbf{Q}(\vec{x}, t), \quad (1.1)$$

ce qui signifie que \mathbf{G} est une fonction \mathbf{Q} du temps t et du point *géométrique* de l'espace renseigné par les coordonnées de \vec{x} .

- la représentation **lagrangienne**, dans laquelle une particule est indexée selon sa position dans la configuration de *référence*. La grandeur est décrite en prenant sa valeur aux positions des *particules* au cours du temps, c'est-à-dire en relevant la valeur de \mathbf{G} le long de leurs *trajectoires* et en calculant leurs *horaires* de parcours :

$$\mathbf{G} = \mathbf{R}(\vec{X}, t), \quad (1.2)$$

ce qui signifie que \mathbf{G} est une fonction \mathbf{R} du temps t et de la position \vec{X} de la *particule* considérée dans sa position de référence. La dérivée d'une grandeur le long d'une trajectoire est appelée dérivée *matérielle* de cette grandeur.

La grandeur \mathbf{G} peut représenter notamment les coordonnées eulériennes de la particule au temps considéré, ce qui nous permet d'écrire :

$$\vec{x}(t) = \vec{x}(\vec{X}, t). \quad (1.3)$$

Au temps de référence t_0 , les deux représentations sont confondues.

Coordonnées d'un point de la membrane

Considérons une membrane décrite par sa configuration de référence \mathcal{C}_0 . Nous choisissons de considérer que, pour la suite, la configuration de référence soit la configuration au temps 0. Nous l'appellerons donc également configuration *initiale*. Dans un repère orthonormé, les coordonnées *lagrangiennes* X_i d'un point P de la membrane seront telles que :

$$\vec{X} = X_i \vec{e}_i, \quad 1 \leq i \leq 3, \quad (1.4)$$

où \vec{X} donne la position du point P dans la configuration de référence. Par la suite, nous n'indiquerons plus les valeurs prises par l'indice i , celui-ci prenant toujours les valeurs 1, 2 et 3.

Au temps t , la membrane va occuper une configuration déformée notée \mathcal{C} . Dans cette configuration les coordonnées *eulériennes* $x_i(t)$ du point P seront telles que :

$$\vec{x}(t) = x_i(\vec{X}, t) \vec{e}_i. \quad (1.5)$$

Nous noterons \vec{u} le vecteur des déplacements entre la configuration de référence et la configuration au temps t :

$$\vec{u}(\vec{x}, t) = \vec{x}(\vec{X}, t) - \vec{X}. \quad (1.6)$$

1.2 Principaux tenseurs de déformation

Si nous prenons un petit vecteur $d\vec{X}$ de la configuration de référence, et que ce vecteur est déformé en un vecteur $d\vec{x}(t)$ au temps t , nous pouvons écrire :

$$d\vec{x}(t) = \frac{\partial \vec{x}(\vec{X}, t)}{\partial \vec{X}} \cdot d\vec{X} = \bar{\bar{F}}(\vec{X}, t) \cdot d\vec{X}. \quad (1.7)$$

Le tenseur $\bar{\bar{F}}$ permet de passer de la configuration de référence \mathcal{C}_0 à la configuration déformée \mathcal{C} (figure 1.1). Il est appelé tenseur des *gradients de déformation*. Remarquons

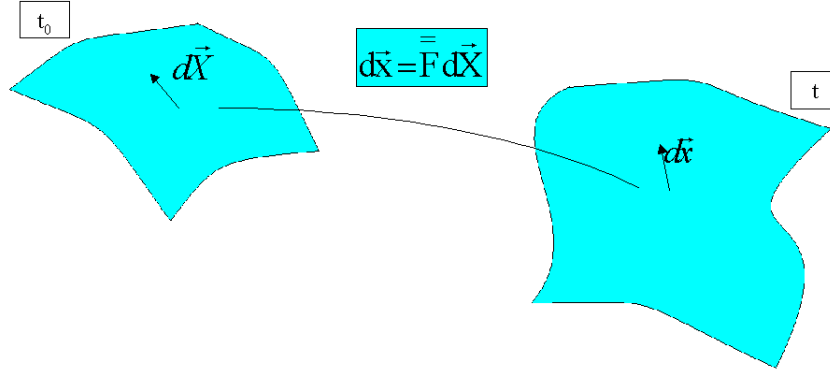


FIG. 1.1 – Illustration d'un lien cinématique entre configurations de référence et déformée.

que

$$\bar{\bar{F}} = \bar{\bar{F}}(\vec{X}, t), \quad (1.8)$$

ce qui signifie bien que la déformation qu'il définit est celle se produisant pour *une* particule, en *un* instant. Il en ira de même pour les autres tenseurs des déformations que nous utiliserons. Cependant, afin d'alléger l'écriture dans la suite du document, nous n'indiquerons pas systématiquement les variables d'espace et de temps dont dépendent les différents tenseurs.

Nous définissons le tenseur des *dilatations* de Cauchy-Green droit $\bar{\bar{C}}(\vec{X}, t)$ par :

$$\bar{\bar{C}}(\vec{X}, t) = \bar{\bar{F}}(\vec{X}, t)^T \cdot \bar{\bar{F}}(\vec{X}, t) \quad (1.9)$$

Celui-ci permet d'obtenir la longueur d'un vecteur déformé :

$$ds^2 = \vec{dx}(t) \cdot \vec{dx}(t) = d\vec{X} \cdot \vec{C}(\vec{X}, t) \cdot d\vec{X}, \quad (1.10)$$

ainsi que la variation d'angle d'un vecteur entre la situation de référence et la configuration déformée :

$$\sin(\theta_{ij}) = \frac{C_{ij}(\vec{X}, t)}{\sqrt{C_{ii}(\vec{X}, t) C_{jj}(\vec{X}, t)}}, \quad (1.11)$$

où θ_{ij} est l'angle formé par les projection de $d\vec{X}$ et de \vec{dx} sur le plan contenant \vec{e}_i et \vec{e}_j .

Pour terminer, nous donnons la définition du tenseur des déformations de Green-Lagrange $\vec{\bar{E}}$:

$$\vec{dx} \cdot \vec{dy} - d\vec{X} \cdot d\vec{Y} = 2 d\vec{X} \cdot \vec{\bar{E}} \cdot d\vec{Y}, \quad (1.12)$$

où \vec{dx} et \vec{dy} sont deux vecteurs de la configuration déformée et $d\vec{X}$ et $d\vec{Y}$ leurs équivalents dans la configuration de référence.

Si $\vec{dx} = \vec{dy}$, le tenseur $\vec{\bar{E}}$ nous donne la différence des carrés des longueurs du vecteur dans les deux configurations :

$$ds^2 - dS^2 = \vec{dx} \cdot \vec{dx} - d\vec{X} \cdot d\vec{X} = 2 d\vec{X} \cdot \vec{\bar{E}} \cdot d\vec{X}, \quad (1.13)$$

où ds et dS sont les longueurs respectives de \vec{dx} et $d\vec{X}$.

En développant l'expression (1.12) à l'aide de (1.7), nous obtenons :

$$d\vec{X} \cdot \vec{\bar{F}}^T \cdot \vec{\bar{F}} \cdot d\vec{Y} - d\vec{X} \cdot d\vec{Y} = 2 d\vec{X} \cdot \vec{\bar{E}} \cdot d\vec{Y}, \quad (1.14)$$

$$d\vec{X} \cdot (\vec{\bar{F}}^T \cdot \vec{\bar{F}} - \vec{\bar{I}}) \cdot d\vec{Y} = 2 d\vec{X} \cdot \vec{\bar{E}} \cdot d\vec{Y}, \quad (1.15)$$

qui nous donne une expression simple de $\vec{\bar{E}}$:

$$\vec{\bar{E}} = \frac{1}{2}(\vec{\bar{C}} - \vec{\bar{I}}). \quad (1.16)$$

1.3 Membranes axisymétriques et membranes 3D

Beaucoup de développements à propos des membranes ont été réalisés pour des structures axisymétriques. Cela permet de travailler avec une théorie, une géométrie et des calculs plus simples. Pour représenter une structure telle que montrée à la figure 1.2, il suffit de prendre une génératrice de la structure et d'étudier le comportement de celle-ci.

Si nous adoptons cette configuration, la géométrie de la membrane n'est décrite que par une seule coordonnée curviligne s :

$$\vec{x} = \vec{x}(s, t). \quad (1.17)$$

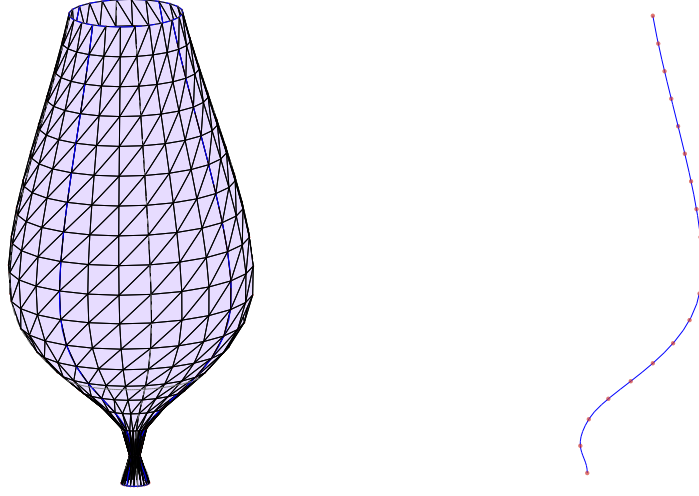


FIG. 1.2 – Représentation 3D d’une membrane axisymétrique, et sa génératrice.

Pour décrire une membrane ne présentant pas de symétrie axiale par contre, il faut considérer une géométrie tridimensionnelle. Dans ce cas, la membrane sera parcourue au moyen de deux coordonnées curvilignes :

$$\vec{x} = \vec{x}(s_1, s_2, t). \quad (1.18)$$

Les calculs à réaliser seront alors beaucoup plus lourds. En contrepartie, l’étude de telles membranes s’étend à des domaines beaucoup plus nombreux et variés. Elle permet de prendre en charge toutes les applications pratiques où une étude est nécessaire.

Dans le cadre de ce mémoire, nous avons choisi de considérer des membranes tridimensionnelles afin de pouvoir coupler notre modèle numérique avec un programme fluide 3D et pour l’aspect plus général de cette approche.

1.4 Bases covariantes et contravariantes

Nous pouvons définir des bases dites *naturelles*, qui suivent l’orientation de la membrane. L’avantage de travailler avec un repère associé à la membrane est que les expressions des principaux tenseurs de déformations sont plus simples et moins gourmandes en temps de calcul que si nous avions travaillé dans un autre repère. Les tenseurs de contraintes verront également leurs expressions simplifiées.

Deux bases naturelles peuvent être adoptées, selon qu’une grandeur s’exprime plus facilement dans l’une ou dans l’autre. Nous distinguons :

- la base **covariante**, dont les deux vecteurs se définissent par :

$$\vec{G}_\alpha(s_1, s_2) = \frac{\partial \vec{X}(s_1, s_2)}{\partial s_\alpha} = \frac{\partial X_i(s_1, s_2)}{\partial s_\alpha} \vec{e}_i, \quad i = 1, 2, 3, \alpha = 1, 2, \quad (1.19)$$

dans la configuration de référence et

$$\vec{g}_\alpha(\vec{X}, t) = \frac{\partial \vec{x}(\vec{X}, t)}{\partial s_\alpha} = \frac{\partial x_i(\vec{X}, t)}{\partial s_\alpha} \vec{e}_i, \quad i = 1, 2, 3, \alpha = 1, 2, \quad (1.20)$$

dans la configuration déformée.

– la base **contravariante**, dont les deux vecteurs se définissent par :

$$\vec{G}^\alpha(s_1, s_2) = \frac{\partial s_\alpha}{\partial X_i(s_1, s_2)} \vec{e}_i, \quad i = 1, 2, 3, \alpha = 1, 2, \quad (1.21)$$

$$\vec{g}^\alpha(\vec{X}, t) = \frac{\partial s_\alpha}{\partial x_i(\vec{X}, t)} \vec{e}_i, \quad i = 1, 2, 3, \alpha = 1, 2, \quad (1.22)$$

respectivement en configurations initiale et déformée.

Ces bases ne sont pas nécessairement orthogonales.

Dans la suite du document, nous noterons par des lettres grecques les indices prenant les valeurs 1 et 2, et par des lettres latines ceux prenant les valeurs 1, 2 et 3.

Pour illustrer ces définitions, considérons une membrane cylindrique (figure 1.3), dont le point P possède les coordonnées suivantes dans la configuration de référence (le raisonnement est similaire pour toute configuration déformée) :

$$\vec{X}(\theta, z) = z \vec{e}_z + R \cos(\theta) \vec{e}_x + R \sin(\theta) \vec{e}_y. \quad (1.23)$$

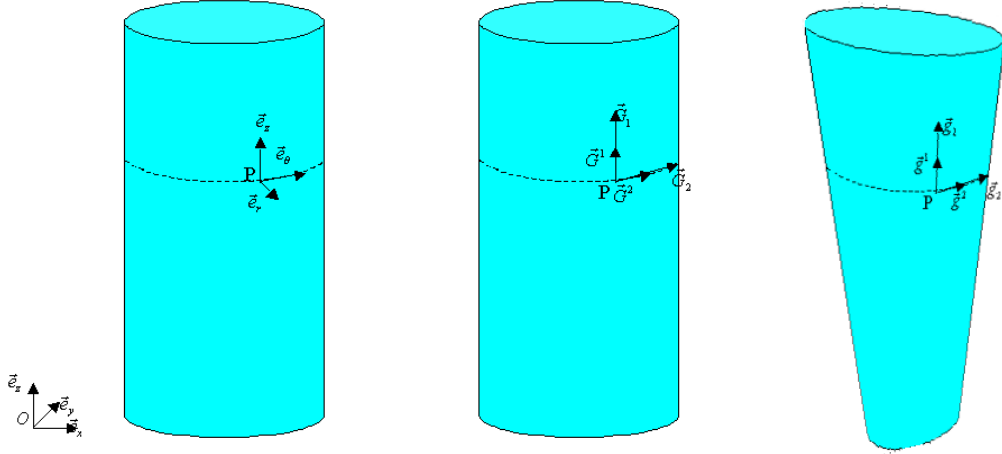


FIG. 1.3 – Illustration de trois types de bases liées à un cylindre : base cylindrique (à gauche), bases naturelles en configuration initiale (au centre) et bases naturelles en configuration déformée (à droite).

Les vecteurs de la base covariante s'obtiennent par :

$$\vec{G}_z = \frac{\partial \vec{X}}{\partial z} = \vec{e}_z, \quad (1.24)$$

$$\vec{G}_\theta = \frac{\partial \vec{X}}{\partial \theta} = -R \sin(\theta) \vec{e}_x + R \cos(\theta) \vec{e}_y. \quad (1.25)$$

Les bases covariantes et contravariantes sont dites *duales*, c'est-à-dire que

$$\vec{G}_\alpha \cdot \vec{G}^\beta = \delta_\alpha^\beta, \quad (1.26)$$

$$\vec{g}_\alpha \cdot \vec{g}^\beta = \delta_\alpha^\beta. \quad (1.27)$$

Cela permet d'obtenir facilement la base contravariante dans l'exemple précédent :

$$\vec{G}^z = \vec{e}_z, \quad (1.28)$$

$$\vec{G}^\theta = -\frac{1}{R} \sin(\theta) \vec{e}_x + \frac{1}{R} \cos(\theta) \vec{e}_y. \quad (1.29)$$

Les métriques de ces bases sont définies par :

$$G_{\alpha\beta} = \vec{G}_\alpha \cdot \vec{G}_\beta, \quad (1.30)$$

$$G^{\alpha\beta} = \vec{G}^\alpha \cdot \vec{G}^\beta, \quad (1.31)$$

$$g_{\alpha\beta} = \vec{g}_\alpha \cdot \vec{g}_\beta, \quad (1.32)$$

$$g^{\alpha\beta} = \vec{g}^\alpha \cdot \vec{g}^\beta, \quad (1.33)$$

et nous pouvons montrer que

$$G_{\alpha\gamma} G^{\gamma\beta} = \delta_\alpha^\beta, \quad (1.34)$$

$$g_{\alpha\gamma} g^{\gamma\beta} = \delta_\alpha^\beta, \quad (1.35)$$

c'est-à-dire que les matrices de ces métriques sont l'inverse l'une de l'autre. Nous pouvons également montrer que ces matrices sont symétriques.

Le passage d'une base à l'autre est défini par les relations :

$$\vec{G}^\alpha = G^{\alpha\beta} \vec{G}_\beta, \quad (1.36)$$

$$\vec{G}_\alpha = G_{\alpha\beta} \vec{G}^\beta, \quad (1.37)$$

$$\vec{g}^\alpha = g^{\alpha\beta} \vec{g}_\beta, \quad (1.38)$$

$$\vec{g}_\alpha = g_{\alpha\beta} \vec{g}^\beta. \quad (1.39)$$

1.5 Tenseurs des déformations en bases locales

Dans les bases naturelles, le tenseur $\bar{\bar{F}}$ du gradient des déformations de la configuration initiale à la configuration déformée s'écrit :

$$\bar{\bar{F}} = \vec{g}_\alpha \otimes \vec{G}^\alpha. \quad (1.40)$$

En effet, si nous prenons les deux vecteurs élémentaires $d\vec{X}$ et $d\vec{x}(t)$ définis au point matériel considéré respectivement sur les configurations initiale et déformée par :

$$d\vec{X} = ds_\alpha \vec{G}_\alpha, \quad (1.41)$$

$$d\vec{x}(t) = ds_\alpha \vec{g}_\alpha(t), \quad (1.42)$$

nous vérifions bien :

$$\bar{\bar{F}} \cdot d\vec{X} = \vec{g}_\alpha \otimes \vec{G}^\alpha \cdot \vec{G}_\gamma ds_\gamma = \vec{g}_\alpha \delta_\alpha^\gamma ds_\gamma = \vec{g}_\alpha ds_\alpha = d\vec{x}, \quad (1.43)$$

où nous définissons un produit scalaire entre un tenseur $\bar{\bar{T}} = T_{\alpha\beta} \vec{u}_\alpha \otimes \vec{v}_\beta$ et un vecteur $\vec{x} = x_\gamma \vec{w}_\gamma$, noté $\bar{\bar{T}} \cdot \vec{x}$, comme :

$$\bar{\bar{T}} \cdot \vec{x} = T_{\alpha\beta} \vec{u}_\alpha (\vec{v}_\beta \cdot \vec{w}_\gamma) x_\gamma. \quad (1.44)$$

Le tenseur $\bar{\bar{F}}$ s'écrit de manière simple dans les bases naturelles initiale et déformée. Il peut néanmoins être utile de se convaincre de sa définition en réécrivant son expression dans la base usuelle :

$$\bar{\bar{F}} = \vec{g}_\alpha \otimes \vec{G}^\alpha = \left(\frac{\partial x_i}{\partial s_\alpha} \vec{e}_i \right) \otimes \left(\frac{\partial s_\alpha}{\partial X_j} \vec{e}_j \right), \quad (1.45)$$

$$= \frac{\partial x_i}{\partial X_j} \vec{e}_i \otimes \vec{e}_j. \quad (1.46)$$

Avec,

$$d\vec{X} = dX_i \vec{e}_i, \quad (1.47)$$

nous avons :

$$\bar{\bar{F}} \cdot d\vec{X} = \frac{\partial x_k}{\partial X_j} \vec{e}_k \otimes \vec{e}_j \cdot \vec{e}_i dX_i, \quad (1.48)$$

$$= \frac{\partial x_k}{\partial X_j} \vec{e}_k \delta_i^j dX_i, \quad (1.49)$$

$$= \frac{\partial x_k}{\partial X_i} dX_i \vec{e}_k, \quad (1.50)$$

$$= dx_k \vec{e}_k, \quad (1.51)$$

$$= d\vec{x}. \quad (1.52)$$

Le tenseur $\bar{\bar{C}}$ des dilatations de Cauchy-Green peut également s'écrire de manière simple. Si nous considérons que :

$$\bar{\bar{F}}^T = \vec{G}^\alpha \otimes \vec{g}_\alpha, \quad (1.53)$$

nous trouvons :

$$\bar{\bar{C}} = \bar{\bar{F}}^T \cdot \bar{\bar{F}}, \quad (1.54)$$

$$= \left(\delta_\alpha^\beta \vec{G}^\alpha \otimes \vec{g}_\beta \right) \cdot \left(\delta_\gamma^\zeta \vec{g}_\gamma \otimes \vec{G}^\zeta \right), \quad (1.55)$$

$$= g_{\beta\gamma} \delta_\alpha^\beta \delta_\gamma^\zeta \vec{G}^\alpha \otimes \vec{G}^\beta, \quad (1.56)$$

$$= g_{\alpha\beta} \vec{G}^\alpha \otimes \vec{G}^\beta. \quad (1.57)$$

où nous définissons un produit scalaire entre deux tenseurs $\bar{\bar{A}} = A_{\alpha\beta} \vec{a}_\alpha \otimes \vec{b}_\beta$ et $\bar{\bar{B}} = B_{\gamma\zeta} \vec{c}_\gamma \otimes \vec{d}_\zeta$, noté $\bar{\bar{A}} \cdot \bar{\bar{B}}$, comme :

$$\bar{\bar{A}} \cdot \bar{\bar{B}} = A_{\alpha\beta} B_{\gamma\zeta} (\vec{b}_\beta \cdot \vec{c}_\gamma) \vec{a}_\alpha \otimes \vec{d}_\zeta. \quad (1.58)$$

Notations : lorsque nous représentons un tenseur par un lettre munie de 2 indices bas, par exemple $A_{\alpha\beta}$, le tenseur correspondant est en fait $\bar{\bar{A}} = A_{\alpha\beta} \vec{G}^\alpha \otimes \vec{G}^\beta$, c'est-à-dire que les $A_{\alpha\beta}$ sont les composantes du tenseur $\bar{\bar{A}}$ dans la base contravariante de la configuration initiale au point considéré. Lorsque nous voulons écrire les composantes d'un tenseur dans la base covariante, nous écrivons $A^{\alpha\beta}$. Ainsi,

$$\bar{\bar{A}} = A_{\alpha\beta} \vec{G}^\alpha \otimes \vec{G}^\beta, \quad (1.59)$$

$$\bar{\bar{B}} = B_\alpha^\beta \vec{G}^\alpha \otimes \vec{G}_\beta, \quad (1.60)$$

$$\bar{\bar{C}} = C_\beta^\alpha \vec{G}_\alpha \otimes \vec{G}^\beta, \quad (1.61)$$

$$\bar{\bar{D}} = D^{\alpha\beta} \vec{G}_\alpha \otimes \vec{G}_\beta. \quad (1.62)$$

2 Petites et grandes déformations

Dans de nombreuses applications, la théorie des petites déformations permet d'établir des modèles relativement proches de la réalité. Lorsque l'on cherche à modéliser un objet dont les déformations sont importantes, une théorie plus générale est nécessaire. Dans cette section, nous montrons que la théorie des petites déformations est une version simplifiée de la théorie des grandes déformations.

Propriétés des petits déplacements

Les petites déformations sont caractérisées par les propriétés suivantes :

- Configurations de référence et déformée très proches.
- Les représentations lagrangiennes et eulériennes de tous les champs sont confondues.
- Les dérivées partielles spatiales de tous les champs s'effectuent indifféremment par rapport aux coordonnées lagrangiennes et eulériennes.
- Les dérivées matérielles de tous les champs sont confondues avec leurs dérivées temporelles.

2.1 Linéarisation des déformations

Le tenseur des gradients de déformation $\bar{\bar{F}}$, défini par l'équation (1.7), peut se réécrire :

$$\bar{\bar{F}} = \frac{\partial \vec{x}}{\partial \vec{X}} = \frac{\partial (\vec{u} + \vec{X})}{\partial \vec{X}} = \frac{\partial \vec{u}}{\partial \vec{X}} + \bar{\bar{I}}. \quad (1.63)$$

En introduisant cette expression dans l'expression (1.16) de $\bar{\bar{E}}$, nous obtenons :

$$\bar{\bar{E}} = \frac{1}{2} \left(\left(\bar{\bar{I}} + \frac{\partial \vec{u}}{\partial \vec{X}} \right)^T \cdot \left(\bar{\bar{I}} + \frac{\partial \vec{u}}{\partial \vec{X}} \right) - \bar{\bar{I}} \right), \quad (1.64)$$

$$= \frac{1}{2} \left(\left(\frac{\partial \vec{u}}{\partial \vec{X}} \right)^T + \left(\frac{\partial \vec{u}}{\partial \vec{X}} \right) + \left(\frac{\partial \vec{u}}{\partial \vec{X}} \right)^T \cdot \left(\frac{\partial \vec{u}}{\partial \vec{X}} \right) \right). \quad (1.65)$$

Nous pouvons remarquer que le tenseur des déformations en grandes déformations contient un terme quadratique, signifiant que les analyses en grandes déformations sont non-linéaires. Lorsque les déformations sont peu importantes, le terme quadratique est négligeable devant les deux autres. Si nous le supprimons, nous nous trouvons dans la théorie des petites déformations, où l'équivalent de $\bar{\bar{E}}$ est le tenseur des déformations infinitésimales $\bar{\epsilon}$:

$$\bar{\epsilon} = \frac{1}{2} \left(\left(\frac{\partial \vec{u}}{\partial \vec{X}} \right)^T + \left(\frac{\partial \vec{u}}{\partial \vec{X}} \right) \right) \approx \frac{1}{2} \left(\left(\frac{\partial \vec{u}}{\partial \vec{x}} \right)^T + \left(\frac{\partial \vec{u}}{\partial \vec{x}} \right) \right). \quad (1.66)$$

Les propriétés des petits déplacements ne sont plus valables en grandes déformations, car elles s'appuient sur la suppression du terme quadratique de l'expression du tenseur des déformations. Il est important de noter également que les dérivées partielles dans le tenseur des déformation de Green-Lagrange $\bar{\bar{E}}$ sont définies par rapport à la configuration initiale. Ce tenseur décrit donc les déformations en coordonnées lagrangiennes.

2.2 Exemple du 1D

La simplification décrite ci-dessus pour passer à la théorie des petites déformations apparaît très clairement si nous considérons une déformation dans un espace à une dimension.

Considérons un segment de longueur l_0 dans la configuration de référence et l au temps t . Le tenseur du gradient des déformations devient un simple scalaire et vaut :

$$F = \frac{l}{l_0}. \quad (1.67)$$

Les déformations de Green-Lagrange deviennent donc :

$$E = \frac{1}{2} (F^2 - 1) = \frac{1}{2} \left(\frac{l^2}{l_0^2} - 1 \right) = \frac{1}{2} \left(\frac{(l_0 + \Delta l)^2}{l_0^2} - 1 \right), \quad (1.68)$$

$$= \frac{\Delta l}{l_0} + \frac{\Delta l^2}{2l_0^2}. \quad (1.69)$$

Si les déformations sont petites, le deuxième terme est négligeable par rapport au premier et nous avons bien :

$$E = \frac{\Delta l}{l_0} = \epsilon. \quad (1.70)$$

3 Comportement du matériau

3.1 Principes et hypothèses

Pour un matériau élastique, les contraintes dérivent d'une énergie de déformation. Nous souhaiterions donc obtenir une loi de comportement permettant de lier les déformations à une énergie W , afin d'en déduire une loi contraintes-déformations.

Cette loi doit être valable pour des matériaux répondant à nos hypothèses, à savoir des matériaux incompressibles, homogènes et isotropes.

La plupart des lois de comportement existantes sont formulées en fonctions des élongations principales ou des invariants du tenseur des dilatations de Cauchy-Green $\bar{\bar{C}}$. En effet, nous nous trouvons dans le cas d'un matériau isotrope ; nous nous attendons donc à ce que l'énergie de déformation ne dépende que de valeurs indépendantes de tout repère.

3.2 Élongations principales et invariants du tenseur des dilatations

Les élongations principales $\lambda_1, \lambda_2, \lambda_3$ d'un matériau déformé sont les racines carrées des trois valeurs propres du tenseur des dilatations de Cauchy-Green. Géométriquement, elles correspondent à l'étirement de la matière dans les trois directions principales.

Les invariants de $\bar{\bar{C}}$ sont au nombre de trois :

$$I_1 = \text{tr } \bar{\bar{C}}, \quad (1.71)$$

$$I_2 = \frac{1}{2} \left[\left(\text{tr } \bar{\bar{C}} \right)^2 - \text{tr } \bar{\bar{C}}^2 \right], \quad (1.72)$$

$$I_3 = \det \bar{\bar{C}}. \quad (1.73)$$

Tout comme les élongations, ceux-ci sont indépendants du repère.

Les invariants peuvent être formulés en fonctions des élongations :

$$I_1 = \lambda_1^2 + \lambda_2^2 + \lambda_3^2, \quad (1.74)$$

$$I_2 = \lambda_1^2 \lambda_2^2 + \lambda_2^2 \lambda_3^2 + \lambda_1^2 \lambda_3^2, \quad (1.75)$$

$$I_3 = \lambda_1^2 \lambda_2^2 \lambda_3^2, \quad (1.76)$$

et inversement :

$$\lambda_k^2 = \frac{1}{3} \left[I_3 + 2 (I_1^2 - 3I_2)^{1/2} \cos \left(\frac{1}{3} (\theta + 2\pi k) \right) \right], \quad k = 1, 2, 3, \quad (1.77)$$

où θ est défini par :

$$\theta = \arccos \left(\frac{2I_1^3 - 9I_1 I_2 + 27I_3}{2(I_1^2 - 3I_2)^{3/2}} \right). \quad (1.78)$$

3.3 Energie de déformation

Nous avons fait l'hypothèse d'un matériau incompressible. Cela implique que

$$I_3 = \lambda_1^2 \lambda_2^2 \lambda_3^2 = 1. \quad (1.79)$$

Physiquement, cela signifie que si nous tirons sur la membrane, non seulement elle s'allonge dans le sens de la traction, mais en plus elle se rétrécit dans la direction perpendiculaire à la membrane.

L'énergie de déformation va donc s'exprimer en fonction de I_1 et I_2 ou de λ_1 et λ_2 .

Une expression répandue pour des matériaux hyperélastiques est la formule de Mooney [7] :

$$W = \sum_{i,j=0}^{\infty} C_{ij} (I_1 - 3)^i (I_2 - 3)^j, \quad (1.80)$$

où les C_{ij} sont les constantes matérielles du modèle.

Plusieurs versions simplifiées de cette expression sont utilisées, la plus courante étant le modèle de Mooney-Rivlin [8] :

$$W = C_1 (I_1 - 3) + C_2 (I_2 - 3), \quad (1.81)$$

où (C_1, C_2) sont les constantes matérielles du modèle. C'est cette expression que nous utiliserons.

Dans la littérature, nous trouvons également les modèles cubique, néo-hookéen, de Isihara, de Gent et Thomas et de Hart-Smith, respectivement écrits :

$$W = C_1 (I_1 - 3) + C_2 (I_1 - 3)^2 + C_3 (I_1 - 3)^3, \quad (1.82)$$

$$W = \frac{1}{2} G (I_1 - 3), \quad (1.83)$$

$$W = C_1 (I_1 - 3) + C_2 (I_1 - 3)^2 + C_3 (I_2 - 3), \quad (1.84)$$

$$W = C_1 (I_1 - 3) + C_2 \ln \left(\frac{I_2}{3} \right), \quad (1.85)$$

$$W = G \left[\int e^{k_1 (I_1 - 3)^2} dI_1 + k_2 \ln \left(\frac{I_2}{3} \right) \right], \quad (1.86)$$

où les grandeurs $(C_1, C_2, C_3, k_1, k_2)$ sont les constantes matérielles des différents modèles.

D'autres modèles utilisent les élongations principales. C'est le cas de celui d'Ogden :

$$W = \sum_{i=1}^N \frac{\mu_i}{\alpha_i} (\lambda_1^{\alpha_i} + \lambda_2^{\alpha_i} + \lambda_3^{\alpha_i} - 3), \quad (1.87)$$

où $(\mu_i, \alpha_i)_{i=1,N}$ sont des constantes matérielles.

Matériaux utilisés

Pour une bonne représentation des éléments examinés, il est important de bien connaître les caractéristiques de leurs matériaux. Pour rappel, nous modélisons des matériaux isotropes, homogènes, incompressibles et hyperélastiques.

Le tableau 1.1 donne quelques exemples de valeurs de C_1 et C_2 trouvées dans la littérature pour l'expression de Mooney-Rivlin :

Matériau		Propriétés		
		C_1 [MPa]	C_2 [MPa]	ϱ [kg/m ³]
Caoutchouc « moyen »	[9]	0.13	0.095	1500
Latex naturel	[10]	0.1184	0.1057	1500
Modèle du myocarde	[11]	2.1	0	1100
Plastic HIPS (145 °C)	[12]	0.085	0	1400
Plastic ABS (145 °C)	[12]	0.23	0	1400

TAB. 1.1 – Valeurs des coefficients de Mooney-Rivlin pour quelques matériaux (HIPS = high impact polystyrene skin).

Une bonne approximation de la constante C_1 consiste à prendre le module de Young du matériau divisé par 6 [4]. Le module est généralement plus facile à trouver que les constantes de Mooney-Rivlin.

3.4 Relation contraintes-déformations

Comme nous l'avons dit plus haut, pour un matériau élastique, les contraintes dérivent d'une énergie de déformation W .

En petite déformation, le tenseur des contraintes de Cauchy peut être défini comme étant la dérivée de l'énergie de déformation par rapport au tenseur des déformations infinitésimales :

$$\bar{\sigma} = \frac{\partial W}{\partial \bar{\epsilon}}. \quad (1.88)$$

Puisque nous travaillons en petites déformations, ce tenseur est défini de la même manière dans la configuration initiale et dans la configuration déformée.

En grandes déformations, les contraintes peuvent être décrites en coordonnées eulériennes ou lagrangiennes, donnant lieu à des expressions cinématiquement différentes. Le tenseur des contraintes de Cauchy $\bar{\sigma}$ donne les contraintes en coordonnées eulériennes, alors que le second tenseur des contraintes de Piola-Kirchhoff $\bar{\bar{S}}$ est défini sur la configuration de référence. La relation cinématique liant $\bar{\bar{S}}$ à $\bar{\sigma}$ est la suivante :

$$\bar{\bar{S}} = \det(\bar{\bar{F}}) \bar{\bar{F}}^T \bar{\sigma} \bar{\bar{F}}. \quad (1.89)$$

Dans notre cas, nous travaillons avec une formulation lagrangienne totale. Nous utilisons donc le tenseur $\bar{\bar{S}}$, qui se définit plus exactement par :

$$\bar{\bar{S}}_{\alpha\beta} = \frac{\partial W}{\partial \bar{\bar{E}}_{\alpha\beta}}. \quad (1.90)$$

La forme caractéristique de la courbe contraintes-déformations d'un matériau hyperélastique est donnée par la figure 1.4. La courbe met en évidence le caractère non-

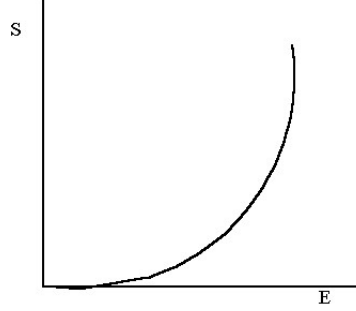


FIG. 1.4 – Relation contraintes-déformations dans un espace à une dimension, S représente les contraintes, E les déformations.

linéaire de la relation.

Pour ces matériaux, $\bar{\bar{S}}$ peut être obtenu facilement en dérivant W par rapport au tenseur des déformations de Green-Lagrange. Nous utiliserons donc l'expression suivante :

$$\bar{\bar{S}} = \frac{\partial W}{\partial \bar{\bar{C}}} \frac{\partial \bar{\bar{C}}}{\partial \bar{\bar{E}}} = 2 \frac{\partial W}{\partial \bar{\bar{C}}}, \quad (1.91)$$

pour laquelle nous avons utilisé (1.16).

Pour un matériau incompressible ($I_3 = 1$), nous pouvons réécrire cette égalité sous la forme suivante :

$$\bar{\bar{S}} = 2 \frac{\partial W}{\partial \bar{\bar{C}}} = 2 \frac{\partial W}{\partial I_i} \frac{\partial I_i}{\partial \bar{\bar{C}}} \Big|_{i=1, 2; I_3=1}, \quad (1.92)$$

où les dérivées $\frac{\partial I_i}{\partial \bar{\bar{C}}}$ s'écrivent :

$$\frac{\partial I_1}{\partial \bar{\bar{C}}} = \bar{\bar{I}}, \quad (1.93)$$

$$\frac{\partial I_2}{\partial \bar{\bar{C}}} = I_1 \cdot \bar{\bar{I}} - \bar{\bar{C}}, \quad (1.94)$$

$$\frac{\partial I_3}{\partial \bar{\bar{C}}} = I_3 \cdot \bar{\bar{C}}^{-1}. \quad (1.95)$$

Dans le cas du modèle de Mooney-Rivlin (équation (1.81)), cette expression devient :

$$\bar{\bar{S}} = 2C_1 \frac{\partial I_1}{\partial \bar{\bar{C}}} + 2C_2 \frac{\partial I_2}{\partial \bar{\bar{C}}}. \quad (1.96)$$

3.5 Réduction à un problème plan

Pour éviter d'avoir à calculer le tenseur $\bar{\bar{C}}$ et ses invariants dans le repère usuel, ce qui nous ferait perdre tout le bénéfice de la cinématique plane, nous allons exprimer ses invariants en fonction des invariants d'un tenseur sur un espace de dimension 2, c'est-à-dire le plan tangent à la membrane au point considéré. C'est ici que les bases covariantes et contravariantes définies plus haut interviennent.

Si nous désignons par $\bar{\bar{F}}'$ le tenseur des gradients de déformation dans ce plan, nous écrivons :

$$\bar{\bar{F}}' = \vec{g}_1 \otimes \vec{G}^1 + \vec{g}_2 \otimes \vec{G}^2. \quad (1.97)$$

La différence avec le tenseur $\bar{\bar{F}}$ est que $\bar{\bar{F}}'$ ne prend pas en compte les déformations sur l'épaisseur de la membrane. Ces déformations peuvent être caractérisées par le rapport $\frac{h}{H}$, où H est l'épaisseur dans la configuration initiale et h est l'épaisseur dans la configuration déformée. Ce rapport est donné par la racine carrée de la troisième valeur propre de $\bar{\bar{C}}$:

$$\lambda_3 = \frac{h}{H}, \quad (1.98)$$

et est supposé uniforme sur l'épaisseur de la membrane.

Nous définissons ensuite $\bar{\bar{C}}'$:

$$\bar{\bar{C}}' = \bar{\bar{F}}'^T \bar{\bar{F}}'. \quad (1.99)$$

Nous noterons I'_1 et I'_2 ses deux invariants,

$$I'_1 = \text{tr} \bar{\bar{C}}', \quad (1.100)$$

$$I'_2 = \det \bar{\bar{C}}'. \quad (1.101)$$

Les valeurs de ces invariants se calculent par :

$$I'_1 = C^1_1 + C^2_2, \quad (1.102)$$

$$I'_2 = C^1_1 C^2_2 - C^1_2 C^2_1, \quad (1.103)$$

en utilisant les valeurs de C^α_β déduites de

$$C^\alpha_\beta = G^{\gamma\alpha} g_{\gamma\beta}, \quad (1.104)$$

puisque

$$\bar{\bar{C}} = g_{\gamma\beta} \vec{G}^\gamma \otimes \vec{G}^\beta, \quad (1.105)$$

$$= g_{\gamma\beta} \left(G^{\gamma\alpha} \vec{G}_\alpha \right) \otimes \vec{G}^\beta, \quad (1.106)$$

$$= C^\alpha_\beta \vec{G}_\alpha \otimes \vec{G}^\beta. \quad (1.107)$$

Nous pouvons écrire les invariants de $\bar{\bar{C}}$ en fonction de I'_1 et I'_2 :

$$I_1 = I'_1 + \frac{1}{I'_2}, \quad (1.108)$$

$$I_2 = I'_2 + \frac{I'_1}{I'_2}, \quad (1.109)$$

$$I_3 = 1. \quad (1.110)$$

L'incompressibilité revient à écrire :

$$I_3 = \det \bar{\bar{C}} = \lambda_3^2 (C_1^1 C_2^2 - C_2^1 C_1^2) = \lambda_3^2 I_2' = 1, \quad (1.111)$$

et donc :

$$\lambda_3^2 = \frac{1}{I_2'}. \quad (1.112)$$

Nous pouvons réécrire le tenseur des contraintes de Piola-Kirchhoff dans le plan de la membrane :

$$\bar{\bar{S}} = 2 \frac{\partial W}{\partial \bar{\bar{C}}} \Big|_{I_3=1} = 2 \frac{\partial W}{\partial \bar{\bar{C}}'} = 2 \frac{\partial W}{\partial I_1} \frac{\partial I_1}{\partial \bar{\bar{C}}'} + 2 \frac{\partial W}{\partial I_2} \frac{\partial I_2}{\partial \bar{\bar{C}}'}. \quad (1.113)$$

Les dérivées des invariants de $\bar{\bar{C}}$ par rapport à $\bar{\bar{C}}'$ sont les suivantes :

$$\frac{\partial I_1}{\partial \bar{\bar{C}}'} = \frac{\partial I_1'}{\partial \bar{\bar{C}}'} - \frac{1}{I_2'^2} \frac{\partial I_2'}{\partial \bar{\bar{C}}'}, \quad (1.114)$$

$$\frac{\partial I_2}{\partial \bar{\bar{C}}'} = \frac{\partial I_2'}{\partial \bar{\bar{C}}'} + \frac{1}{I_2'} \frac{\partial I_1'}{\partial \bar{\bar{C}}'} - \frac{I_1'}{I_2'^2} \frac{\partial I_2'}{\partial \bar{\bar{C}}'}, \quad (1.115)$$

où les dérivées des invariants de $\bar{\bar{C}}'$ par rapport à $\bar{\bar{C}}'$ sont :

$$\frac{\partial I_1'}{\partial \bar{\bar{C}}'} = \bar{\bar{I}}, \quad (1.116)$$

$$\frac{\partial I_2'}{\partial \bar{\bar{C}}'} = I_1' \bar{\bar{I}} - \bar{\bar{C}}'. \quad (1.117)$$

Les dérivées de I_1 et I_2 peuvent être écrites dans les bases naturelles :

$$\frac{\partial I_1}{\partial \bar{\bar{C}}'} = \left[\left(1 - \frac{I_1'}{I_2'^2} \right) G^{\alpha\beta} + \frac{1}{I_2'^2} g^{\alpha\beta} \right] \vec{G}_\alpha \otimes \vec{G}_\beta, \quad (1.118)$$

$$\frac{\partial I_2}{\partial \bar{\bar{C}}'} = \left[\left(I_1' + \frac{1}{I_2'} - \frac{I_1'^2}{I_2'^2} \right) G^{\alpha\beta} + \left(\frac{I_1'}{I_2'^2} - 1 \right) g^{\alpha\beta} \right] \vec{G}_\alpha \otimes \vec{G}_\beta. \quad (1.119)$$

En remplaçant dans l'équation (1.113) $\frac{\partial I_1}{\partial \bar{\bar{C}}'}$ et $\frac{\partial I_2}{\partial \bar{\bar{C}}'}$ par leurs expressions dans (1.118) et (1.119), nous obtenons :

$$\begin{aligned} \bar{\bar{S}} = & 2 \frac{\partial W}{\partial I_1} \left[\left(1 - \frac{I_1'}{I_2'^2} \right) G^{\alpha\beta} + \frac{1}{I_2'^2} g^{\alpha\beta} \right] \vec{G}_\alpha \otimes \vec{G}_\beta, \\ & + 2 \frac{\partial W}{\partial I_2} \left[\left(I_1' + \frac{1}{I_2'} - \frac{I_1'^2}{I_2'^2} \right) G^{\alpha\beta} + \left(\frac{I_1'}{I_2'^2} - 1 \right) g^{\alpha\beta} \right] \vec{G}_\alpha \otimes \vec{G}_\beta, \end{aligned} \quad (1.120)$$

ou, en introduisant le modèle de Mooney-Rivlin pour W (équation (1.81)) :

$$\begin{aligned} \bar{\bar{S}} = & 2C_1 \left[\left(1 - \frac{I_1'}{I_2'^2} \right) G^{\alpha\beta} + \frac{1}{I_2'^2} g^{\alpha\beta} \right] \vec{G}_\alpha \otimes \vec{G}_\beta, \\ & + 2C_2 \left[\left(I_1' + \frac{1}{I_2'} - \frac{I_1'^2}{I_2'^2} \right) G^{\alpha\beta} + \left(\frac{I_1'}{I_2'^2} - 1 \right) g^{\alpha\beta} \right] \vec{G}_\alpha \otimes \vec{G}_\beta. \end{aligned} \quad (1.121)$$

Cette relation exprime le tenseur des contraintes $\bar{\bar{S}}$ comme une fonction des invariants du tenseur plan $\bar{\bar{C}}'$ et des métriques des bases naturelles. Cela nous permet d'obtenir d'une manière relativement peu coûteuse les contraintes en fonction de la déformation.

Chapitre 2

Méthodes de discrétisation

Dans cette section, nous exposons les principaux outils de discrétisation spatiale et temporelle dont nous aurons besoin pour la réalisation complète d'un code de calcul, et nous montrons comment appliquer la théorie vue précédemment au système discret obtenu. La méthode des éléments finis nous permettra de créer une discrétisation spatiale de la membrane, tandis que deux schémas temporels, un explicite et un implicite, seront étudiés.

1 Méthode des éléments finis

La méthode des éléments finis est un des outils les plus efficaces et les plus généraux de la simulation numérique. Elle s'applique aux problèmes de mécanique des milieux continus comprenant :

- les problèmes de mécanique des solides : élasticité, plasticité, mise en forme ;
- les problèmes de mécanique des fluides, d'hydraulique ;
- les problèmes de transfert de chaleur ;
- les problèmes d'électromagnétisme ;
- ...

1.1 Démarche des éléments finis

D'un point de vue mathématique, la méthode des éléments finis permet de calculer une approximation d'une fonction qui vérifie un système d'équations aux dérivées partielles à l'intérieur d'un domaine, et un ensemble de conditions aux limites sur les frontières de ce domaine.

Cet outil est basé sur une *discrétisation* du domaine dans lequel nous désirons effectuer la simulation. Celui-ci est découpé en un nombre plus ou moins grand d'*éléments*. A l'intérieur de chacun de ces éléments, un ou plusieurs points sont privilégiés et constituent les *nœuds* de la discrétisation. C'est en ces nœuds que seront appliquées

les équations du système.

La solution est formulée comme une somme de fonctions de base appelées *fonctions de forme globales* pondérées par les valeurs de la solution approchée aux nœuds (figure 2.1 pour un domaine unidimensionnel) :

$$u^h(\vec{x}) = \sum_{i=1}^N U_i \tau_i(\vec{x}), \quad (2.1)$$

où N est le nombre de nœuds, $u^h(\vec{x})$ est la solution approchée, U_i est la solution au nœud i , et τ_i est la fonction de forme associée au nœud i .

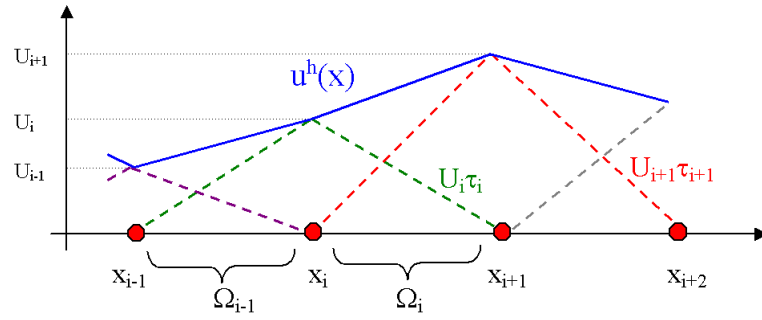


FIG. 2.1 – Approximation d'une fonction $u(x)$ par une combinaison linéaire de valeurs nodales et de fonctions de forme globales.

Chacune de ces fonctions est nulle sur l'entièreté du domaine, excepté sur quelques éléments contigus. La solution sur un élément est donc la somme d'un nombre limité de fonctions. Les morceaux de fonctions de forme non nulles sur un élément sont appelées fonctions de forme *locales* ϕ_i . La figure 2.2 montre les fonctions de forme globales et

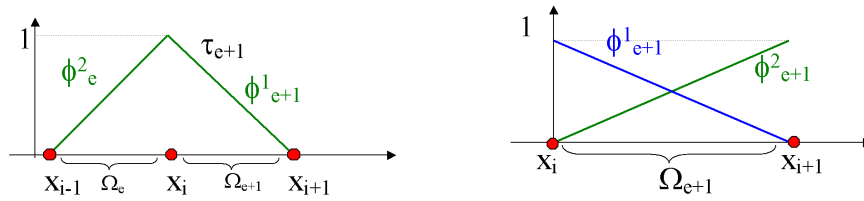


FIG. 2.2 – Illustration en 1D de la fonction de forme globale linéaire associée au nœud X_i (à gauche), et des fonctions de forme locales linéaires sur un élément 1D (à droite).

locales linéaires sur un élément d'un maillage 1D. Pour celles-ci, nous pouvons écrire :

$$\begin{aligned} \tau_e(x) &= \phi_1^e(x), & x &\in \Omega_e, \\ \tau_{e+1}(x) &= \phi_2^e(x), & x &\in \Omega_e, \\ \tau_f(x) &= 0, & f &\notin \{e, e+1\} \text{ et } x \in \Omega_e. \end{aligned}$$

1.2 Éléments parents

Pour commencer, nous souhaitons mettre en évidence la distinction existant entre un nœud et un sommet : un sommet est un point à l'intersection de plusieurs éléments finis, tandis qu'un nœud est un point où est définie une valeur discrète ou nodale de la fonction recherchée. Si nous prenons l'exemple 1D de la figure 2.3, les sommets

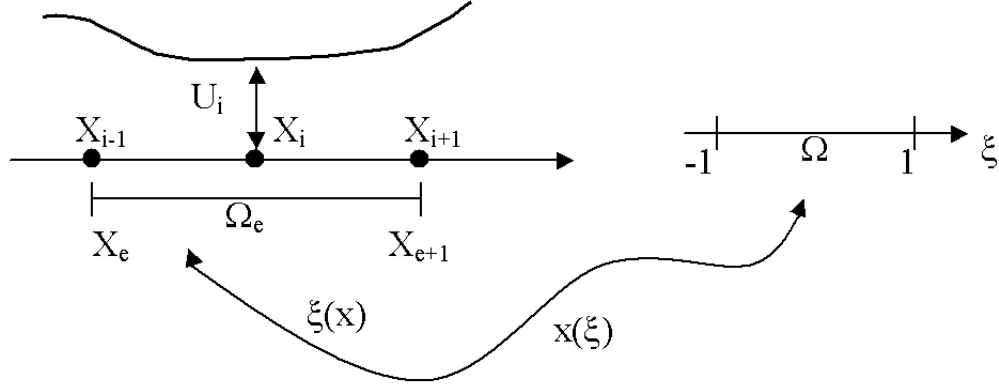


FIG. 2.3 – Isomorphisme entre un élément et l'élément parent en 1D.

sont représentés par les points X_e et X_{e+1} et les nœuds par X_{i-1} , X_i et X_{i+1} . A chaque nœud correspond une valeur U_i , et dans ce cas-ci les sommets de l'élément sont confondus avec les nœuds X_{i-1} et X_{i+1} .

L'expression des fonctions de forme locales est habituellement écrite sur un *élément parent* $\hat{\Omega}$. Il s'agit d'un élément à la géométrie la plus simple possible avec lequel il soit permis d'établir un isomorphisme entre élément et élément parent. La connaissance des caractéristiques de cette transformation permet d'effectuer une partie du travail sur la géométrie simple de l'élément parent et de traduire les résultats sur l'élément quelconque.

En reprenant la figure 2.3, nous pouvons établir un isomorphisme entre un point d'un élément du domaine et un point d'un élément parent, intervalle ouvert $\hat{\Omega} =]-1, +1[$, défini par :

$$x(\xi) = \xi \frac{(X_{e+1} - X_e)}{2} + \frac{(X_{e+1} + X_e)}{2}, \quad (2.2)$$

$$\xi(x) = \frac{2x - (X_{e+1} + X_e)}{(X_{e+1} - X_e)}. \quad (2.3)$$

Les éléments parents que nous utilisons sont décrits par la figure 2.4 représentant deux géométries couramment employées pour des éléments parents en 2D.

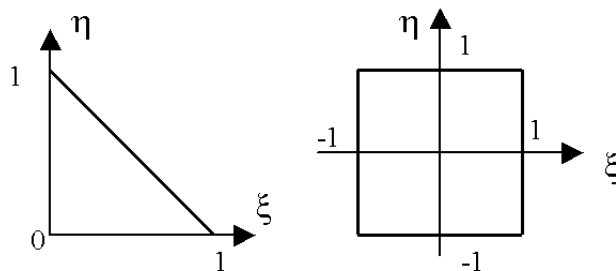


FIG. 2.4 – Eléments parents d'un triangle et d'un quadrangle.

1.3 Eléments utilisés

Maillages de triangles et de quadrangles

Le maillage d'une structure plane peut se faire le plus simplement en décomposant la surface en mailles triangulaires ou en mailles quadrangulaires (figure 2.5). A priori, le choix de quadrangles permet d'éviter certaines difficultés liées à la continuité des dérivées de la solution aux nœuds. En revanche, la réalisation d'un maillage quadrangulaire d'une surface présentant des frontières irrégulières peut s'avérer très difficile.

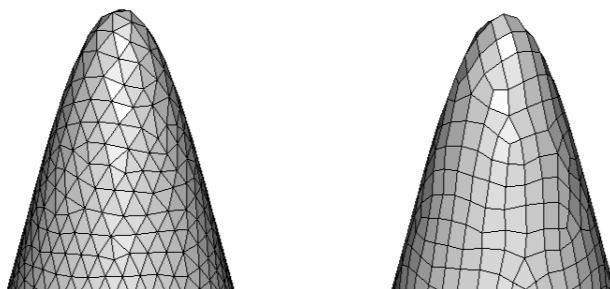


FIG. 2.5 – Exemples de maillages de triangles et de quadrangles.

Les maillages triangulaires, au contraire, s'adaptent facilement à n'importe quelle géométrie. Il est néanmoins très difficile d'établir une correspondance entre les différentes dérivées de la solution aux frontières entre éléments triangulaires. Les fonctions de forme seront également différentes selon le type de mailles utilisées.

Dès le départ, nous avons pris l'option d'adopter des mailles triangulaires afin de faciliter le couplage avec le programme fluide. En effet, celui-ci fonctionne avec des éléments tétraédriques, ce qui nous permettra d'établir une correspondance nœud à nœud entre un maillage surfacique de triangles et un maillage 3D. Nous évitons le problème de la continuité des dérivées aux nœuds en admettant une solution non « lisse », c'est-à-dire présentant des changements de pente d'un élément à l'autre.

Nous avons toutefois implémenté une version du programme utilisant des qua-

drangles. Le programme de calcul de membranes permet finalement de traiter des membranes discrétisées de manière mixte, triangles/quadrangles.

Maillages structurés et non structurés

Les maillages peuvent être répartis en deux catégories (figure 2.6) : les maillages

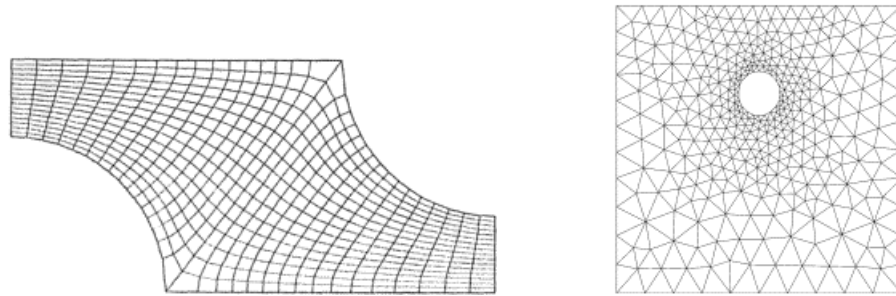


FIG. 2.6 – A gauche, maillage structuré ; à droite, maillage non structuré (figures extraites de [13]).

structurés, qui offrent l'avantage d'une grille de calcul simple et facile à mettre en œuvre, et les maillages non structurés, qui s'adaptent à toutes les géométries, même complexes, et permettent un raffinement du nombre de points de calcul, et donc une meilleure précision dans les résultats du modèle, en fonction des zones d'intérêt. Un maillage structuré consiste donc à mailler régulièrement un domaine selon un principe de grille.

Nous avons choisi de baser notre code sur l'utilisation de maillages non structurés.

Maillages conformes et non conformes

La conformité d'un maillage est définie par plusieurs caractéristiques [14] :

- l'intersection de deux éléments du maillage est soit vide, soit réduite à un sommet ou à une arête complète pour les deux éléments (ou encore à une face complète en 3D),
- tout élément du maillage possède une aire non nulle,
- l'union des éléments du maillage recouvre l'ensemble du domaine.

Les figures 2.7 et 2.8 illustrent ce concept. Sur la deuxième figure, le maillage de droite possède des éléments triangulaires qui servent à faire le lien entre les différents éléments du maillage rectangulaire, rendant celui-ci conforme. Cette technique est souvent utilisée dans le raffinement de maillages [16]. Dans notre cas, tous les maillages utilisés sont des maillages conformes.

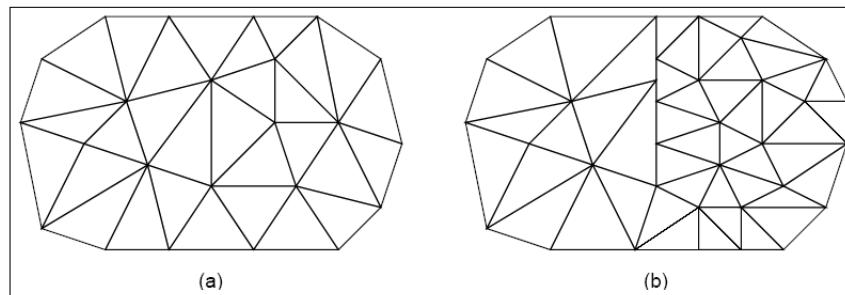


FIG. 2.7 – Maillages conforme (à gauche) et non conforme (à droite) (figures extraites de [15]).

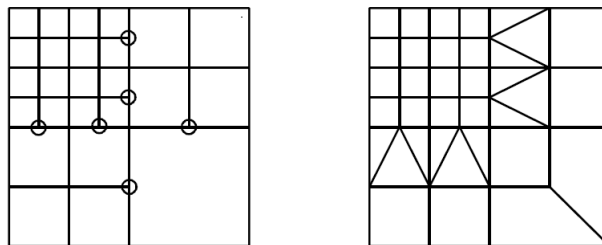


FIG. 2.8 – Maillages non conforme (à gauche) et conforme (à droite) (figures extraites de [16]).

Fonctions de forme

Pour les éléments triangulaires, nous utilisons des fonctions de forme linéaires (figure 2.9). Leurs expressions sur l'élément parent sont :

$$\begin{aligned}\phi_1(s_1, s_2) &= 1 - s_1 - s_2, \\ \phi_2(s_1, s_2) &= s_1, \\ \phi_3(s_1, s_2) &= s_2.\end{aligned}\tag{2.4}$$

Ces fonctions sont les plus simples pour obtenir une solution continue sans rechercher la continuité des pentes entre deux éléments.

Pour l'implémentation de la version par quadrangles, nous utilisons des fonctions

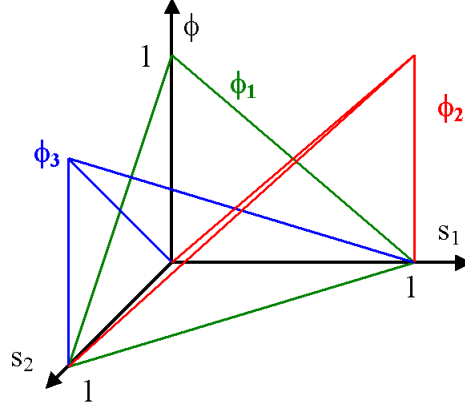


FIG. 2.9 – Fonctions de forme linéaires sur un triangle parent.

de forme bilinéaires, définies sur l'élément parent (figure 2.10) par :

$$\begin{aligned}
 \phi_1(s_1, s_2) &= \frac{(1 - s_1)(1 - s_2)}{4}, \\
 \phi_2(s_1, s_2) &= \frac{(1 + s_1)(1 - s_2)}{4}, \\
 \phi_3(s_1, s_2) &= \frac{(1 + s_1)(1 + s_2)}{4}, \\
 \phi_4(s_1, s_2) &= \frac{(1 - s_1)(1 + s_2)}{4}.
 \end{aligned} \tag{2.5}$$

1.4 Discrétisation des grandeurs principales

Comme expliqué plus haut, nous approximations la position d'un point quelconque de la membrane à l'aide de fonctions de forme définies sur l'élément parent, dans le plan des coordonnées curvilignes (s_1, s_2) . Dans la configuration initiale, nous écrivons ainsi :

$$\mathbf{X}(s_1, s_2) = \mathbf{N}(s_1, s_2) \cdot \mathbf{X}^e, \tag{2.6}$$

où $\mathbf{X}(s_1, s_2)$ est une matrice colonne reprenant les trois coordonnées du point dans la configuration initiale, \mathbf{X}^e est une matrice colonne de longueur $3n^e$ composée des trois coordonnées des n^e nœuds de l'élément e dans la configuration initiale, et $\mathbf{N}(s_1, s_2)$ est une matrice $3 \times 3n^e$ dont les entrées sont les valeurs des fonctions de forme aux coordonnées (s_1, s_2) . Pour des éléments triangulaires à trois nœuds,

$$\mathbf{N} = \begin{bmatrix} \phi_1 & 0 & 0 & \phi_2 & 0 & 0 & \phi_3 & 0 & 0 \\ 0 & \phi_1 & 0 & 0 & \phi_2 & 0 & 0 & \phi_3 & 0 \\ 0 & 0 & \phi_1 & 0 & 0 & \phi_2 & 0 & 0 & \phi_3 \end{bmatrix}. \tag{2.7}$$

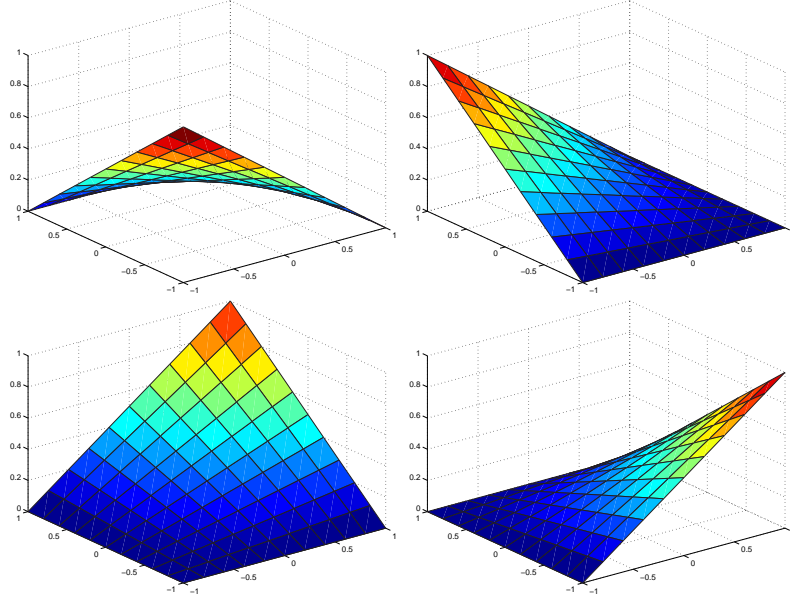


FIG. 2.10 – Fonctions de forme bilinéaires sur un quadrangle parent.

De même, pour le déplacement et la position d'un point dans la configuration déformée :

$$\mathbf{u}(s_1, s_2) = \mathbf{N}(s_1, s_2) \cdot \mathbf{U}^e, \quad (2.8)$$

$$\mathbf{x}(s_1, s_2) = \mathbf{X}(s_1, s_2) + \mathbf{u}(s_1, s_2) = \mathbf{N}(s_1, s_2) \cdot (\mathbf{X}^e + \mathbf{U}^e) = \mathbf{N}(s_1, s_2) \cdot \mathbf{x}^e. \quad (2.9)$$

La vitesse et l'accélération peuvent également être discrétisées par :

$$\dot{\mathbf{u}}(s_1, s_2) = \mathbf{N}(s_1, s_2) \cdot \dot{\mathbf{U}}^e, \quad (2.10)$$

$$\ddot{\mathbf{u}}(s_1, s_2) = \mathbf{N}(s_1, s_2) \cdot \ddot{\mathbf{U}}^e. \quad (2.11)$$

Nous approximations les bases covariantes des configurations initiales et déformées par :

$$\mathbf{G}_\alpha(s_1, s_2) = \frac{\partial \mathbf{N}(s_1, s_2)}{\partial s_\alpha} \cdot \mathbf{X}^e = \mathbf{N}_{,\alpha} \cdot \mathbf{X}^e, \quad (2.12)$$

$$\mathbf{g}_\alpha(s_1, s_2) = \frac{\partial \mathbf{N}(s_1, s_2)}{\partial s_\alpha} \cdot \mathbf{x}^e = \mathbf{N}_{,\alpha} \cdot \mathbf{x}^e. \quad (2.13)$$

2 Calcul des forces internes et externes

Les notions développées précédemment, et notamment la relation contraintes-déformations, vont nous permettre de calculer les forces « de rappel » exercées par un élément en chacun de ses nœuds du fait de sa déformation. Pour ce faire, nous aurons recours au Principe des Travaux Virtuels (PTV).

Les efforts en chacun des nœuds dus aux forces extérieures appliquées à la membrane seront également calculés en partant du PTV.

2.1 Principe des Travaux Virtuels

Le Principe des Travaux Virtuels en dynamique exprime un équilibre, sous un champ \vec{u} de déplacements admissibles, dans lequel la différence des travaux générés par les efforts internes et externes correspond au travail nécessaire pour provoquer le champ des accélérations $\vec{\ddot{u}}$, et ce pour tout champ de déplacements virtuels $\delta\vec{u}$ s'annulant aux frontières.

En intégrant ces travaux sur l'épaisseur H de la membrane et en négligeant les forces d'inertie et les forces volumiques, le Principe des Travaux Virtuels peut s'écrire :

$$g(\vec{u}, \delta\vec{u}, p) = \int_{S_0} H \delta\bar{\bar{E}} : \bar{\bar{S}} dS_0 - \int_S p \delta\vec{u} \cdot \vec{n} dS = \int_{S_0} \delta\vec{u} \cdot (\rho H \vec{\ddot{u}}) dS_0, \quad \forall \delta\vec{u}, \quad (2.14)$$

où \vec{u} est le champ de déplacements, $\vec{\ddot{u}}$ le champ des accélérations, $\delta\vec{u}$ est un champ de déplacements virtuels, ρ , H et S_0 sont la masse volumique, l'épaisseur et la surface de la membrane dans sa configuration initiale, S sa surface en configuration déformée, \vec{n} le vecteur unitaire normal à la membrane déformée, p la pression appliquée sur la membrane, $\bar{\bar{S}}$ le second tenseur des contraintes de Piola-Kirchhoff, et $\bar{\bar{E}}$ est le tenseur des déformations de Green-Lagrange.

Dans cette équation, le premier terme est celui du travail virtuel des forces internes tandis que le second représente le travail virtuel des forces externes.

Moyennant une décomposition de la membrane par éléments finis, cette équation peut être réécrite sous la forme :

$$g(\mathbf{U}, \delta\mathbf{U}, p) = \delta\mathbf{U}^T \cdot [\mathbf{F}_{\text{int}}(\mathbf{U}) - \mathbf{F}_{\text{ext}}(\mathbf{U}, p)] = \delta\mathbf{U}^T \cdot \mathbf{M} \cdot \ddot{\mathbf{U}}, \quad \forall \delta\mathbf{U}, \quad (2.15)$$

où \mathbf{U} , $\ddot{\mathbf{U}}$, $\delta\mathbf{U}$, \mathbf{F}_{int} et \mathbf{F}_{ext} sont les matrices des valeurs nodales de \vec{u} , $\vec{\ddot{u}}$, $\delta\vec{u}$ et des forces internes et externes, et \mathbf{M} est la matrice de masse. Toutes ces matrices, à l'exception de la matrice de masse, sont de taille $n \times 3$, n étant le nombre total de nœuds. Chacune des colonnes (au nombre de trois) de ces matrices indique une composante du vecteur correspondant dans la base orthonormée classique. La matrice de masse est de taille $n \times n$.

2.2 Forces internes

Le terme correspondant aux forces internes pour un élément e dans le Principe des Travaux Virtuels s'écrit :

$$\delta w_{int}^e = \int_{S_0^e} H \delta \mathbf{E} : \mathbf{S} dS_0. \quad (2.16)$$

La partie tensorielle de cette expression s'explicité comme suit :

$$\delta \mathbf{E} : \mathbf{S} = \delta E_{\alpha\beta} S^{\alpha\beta}, \quad (2.17)$$

que nous calculons grâce à

$$E_{\alpha\beta} = \frac{1}{2}(\mathbf{g}_\alpha^T \cdot \mathbf{g}_\beta - \mathbf{G}_\alpha^T \cdot \mathbf{G}_\beta). \quad (2.18)$$

Nous rappelons que les indices α et β sont utilisés pour indiquer qu'ils ne prennent que les valeurs 1 et 2.

Puisque \mathbf{G}_α est indépendant des déplacements, nous avons

$$\delta E_{\alpha\beta} = \frac{1}{2}(\delta \mathbf{g}_\alpha^T \cdot \mathbf{g}_\beta + \delta \mathbf{g}_\beta^T \cdot \mathbf{g}_\alpha), \quad (2.19)$$

avec

$$\delta \mathbf{g}_\alpha = \frac{\partial \mathbf{g}_\alpha}{\partial \mathbf{U}^e} \cdot \delta \mathbf{U}^e = \mathbf{N}_{,\alpha} \cdot \delta \mathbf{U}^e. \quad (2.20)$$

En posant

$$\mathbf{B}_{\alpha\beta} = \frac{1}{2}(\mathbf{g}_\alpha^T \cdot \mathbf{N}_{,\beta} + \mathbf{g}_\beta^T \cdot \mathbf{N}_{,\alpha}), \quad (2.21)$$

nous pouvons écrire

$$\delta E_{\alpha\beta} = \delta \mathbf{U}^{eT} \cdot (\mathbf{B}_{\alpha\beta})^T. \quad (2.22)$$

Enfin, en calculant l'élément de surface dS_0 par

$$dS_0 = \|\mathbf{G}_1 \wedge \mathbf{G}_2\| ds_1 ds_2, \quad (2.23)$$

nous réécrivons les travaux internes

$$\delta w_{int}^e = \delta \mathbf{U}^{eT} \cdot \int_{\Omega} H (\mathbf{B}_{\alpha\beta})^T S^{\beta\alpha} \|\mathbf{G}_1 \wedge \mathbf{G}_2\| ds_1 ds_2 = \delta \mathbf{U}^{eT} \cdot \mathbf{F}_{int}^e, \quad (2.24)$$

où Ω est la surface de l'élément parent.

Nous pouvons donc finalement écrire les forces internes s'exerçant aux nœuds de l'élément e :

$$\mathbf{F}_{int}^e = \int_{\Omega} H (\mathbf{B}_{\alpha\beta})^T S^{\beta\alpha} \|\mathbf{G}_1 \wedge \mathbf{G}_2\| ds_1 ds_2. \quad (2.25)$$

Le terme \mathbf{B} est une matrice d'ordre 4 et de taille $2 \times 2 \times 3n^e \times 1$, un élément $\mathbf{B}_{\alpha\beta}$ de cette matrice étant une matrice de taille $3n^e \times 1$; la matrice des composantes de $\bar{\bar{\mathbf{S}}}$ est de taille (2×2) ; le facteur $\|\mathbf{G}_1 \wedge \mathbf{G}_2\|$ est un scalaire. La règle des indices muets est d'application, c'est-à-dire que lorsque le même indice apparaît deux fois exactement dans une expression, l'expression vaut la somme des évaluations de celle-ci à toutes les valeurs que peut prendre l'indice. Dans ce cas, l'expression résulte bien en une matrice de taille $3n^e \times 1$, dont chaque groupe de trois éléments représente les forces internes appliquées à un nœud dans les trois directions $(\vec{e}_x, \vec{e}_y, \vec{e}_z)$ de l'espace.

2.3 Forces externes

Dans l'équation (2.14) du Principe des Travaux Virtuels, le terme représentant le travail virtuel des forces externes s'écrit, pour l'élément e :

$$\delta w_{ext}^e = \int_{S^e} p \delta \mathbf{u}^T \cdot \mathbf{n} dS. \quad (2.26)$$

En explicitant les différents membres par

$$dS = \| \mathbf{g}_1 \wedge \mathbf{g}_2 \| ds_1 ds_2, \quad (2.27)$$

$$\mathbf{n} = \frac{\mathbf{g}_1 \wedge \mathbf{g}_2}{\| \mathbf{g}_1 \wedge \mathbf{g}_2 \|}, \quad (2.28)$$

$$\delta \mathbf{u} = \mathbf{N} \cdot \delta \mathbf{U}^e, \quad (2.29)$$

nous réécrivons les travaux extérieurs virtuels par

$$\delta w_{ext}^e = \delta \mathbf{U}^{eT} \cdot \int_{\Omega} p \mathbf{N}^T \cdot (\mathbf{g}_1 \wedge \mathbf{g}_2) ds_1 ds_2 = \delta \mathbf{U}^{eT} \cdot \mathbf{F}_{ext}^e, \quad (2.30)$$

ce qui nous permet d'écrire les forces extérieures aux nœuds de l'élément e :

$$\mathbf{F}_{ext}^e = \int_{\Omega} p \mathbf{N}^T \cdot (\mathbf{g}_1 \wedge \mathbf{g}_2) ds_1 ds_2. \quad (2.31)$$

2.4 Matrice de masse

Reprenons le membre de droite de l'équation (2.14) :

$$\int_{S_0} \delta \vec{u} \cdot (\rho H \vec{\ddot{u}}) dS_0, \quad (2.32)$$

ou, après discrétisation, et pour un élément :

$$\int_{S_0^e} \delta \mathbf{u}^T \cdot (\rho H \ddot{\mathbf{u}}) dS_0. \quad (2.33)$$

En utilisant (2.11), (2.23) et (2.29), cette expression se réécrit :

$$\int_{S_0^e} \delta \mathbf{U}^{eT} \cdot \mathbf{N}^T \cdot (\rho H \mathbf{N} \cdot \ddot{\mathbf{U}}) \| \mathbf{G}_1 \wedge \mathbf{G}_2 \| ds_1 ds_2 = \delta \mathbf{U}^{eT} \cdot \mathbf{M} \cdot \ddot{\mathbf{U}}. \quad (2.34)$$

La matrice de masse de l'élément se calcule donc par :

$$\mathbf{M} = \int_{S_0^e} \mathbf{N}^T \cdot (\rho H \| \mathbf{G}_1 \wedge \mathbf{G}_2 \| \mathbf{N}) ds_1 ds_2. \quad (2.35)$$

Les matrices globales des forces et de la masse s'obtiennent en assemblant les matrices locales ainsi obtenues.

2.5 Intégration numérique

Pour intégrer numériquement les forces internes et externes des expressions (2.25) et (2.31), ainsi que la matrice de masse, nous avons recours à la technique de Hammer pour les triangles, et à la technique de Gauss-Legendre pour les quadrangles. Ces techniques consistent à évaluer la fonction à intégrer en un certain nombre de points et à additionner les résultats affectés de coefficients pondérateurs. La technique décrit la position de ces points ainsi que le poids associé à chacun de ceux-ci. Le choix du nombre de points d'intégration dépend du degré du polynôme que nous voudrions pouvoir intégrer exactement.

Technique de Hammer

Les coordonnées des points d'intégration sont données en terme de coordonnées triangulaires $(\Delta_1, \Delta_2, \Delta_3)$. Si nous divisons l'élément en trois triangles en joignant le point de coordonnées $(\Delta_1, \Delta_2, \Delta_3)$ à chacun des sommets, Δ_i sera égal au rapport de l'aire du triangle opposé au sommet i à l'aire totale.

Le rapport entre les coordonnées triangulaires et les coordonnées (s_1, s_2) de l'élément parent s'exprime par :

$$\Delta_1(s_1, s_2) = 1 - s_1 - s_2, \quad (2.36)$$

$$\Delta_2(s_1, s_2) = s_1, \quad (2.37)$$

$$\Delta_3(s_1, s_2) = s_2. \quad (2.38)$$

La littérature fournit les coordonnées triangulaires des points d'intégration de Hammer ainsi que leur poids respectif (ω). Pour le calcul des forces internes, seules les dérivées des fonctions de forme interviennent et celles-ci sont constantes (fonctions de forme linéaires). Nous n'avons donc besoin que d'un seul point d'intégration, choisi de manière quelconque.

Le terme des forces extérieures nécessite seulement l'intégration d'un polynôme linéaire, donc une intégration de Hammer d'ordre 1 (un point) :

Ordre	Δ_1	Δ_2	Δ_3	ω
1	1/3	1/3	1/3	1/2

Une intégration d'ordre 2 (trois points) sera nécessaire lorsque nous prendrons en compte des pressions non constantes sur un élément (voir seconde partie) :

Ordre	Δ_1	Δ_2	Δ_3	ω
2	1/2	1/2	0	1/6
	0	1/2	1/2	1/6
	1/2	0	1/2	1/6

Et enfin, nous utiliserons une intégration d'ordre élevé pour calculer la matrice de masse du système. En effet, son calcul nécessitera l'intégration des produits des fonctions de forme, et un ordre d'intégration plus élevé sera requis. Voici les coordonnées et les poids de l'intégration de Hammer d'ordre 5 (sept points) :

Ordre	Δ_1	Δ_2	Δ_3	ω
5	0.33333	0.33333	0.33333	0.11250
	0.05972	0.47014	0.47014	0.06620
	0.47014	0.05972	0.47014	0.06620
	0.47014	0.47014	0.05972	0.06620
	0.79743	0.10129	0.10129	0.06297
	0.10129	0.79743	0.10129	0.06297
	0.10129	0.10129	0.79743	0.06297

Nous pourrions nous passer d'une intégration si précise et sacrifier un peu de précision pour du temps de calcul, mais la matrice de masse n'est calculée qu'une fois au cours de l'exécution. Le désavantage en temps est donc négligeable.

Technique de Gauss-Legendre

L'intégrale de Gauss-Legendre (sur l'élément parent) s'écrit :

$$\int_{-1}^1 \int_{-1}^1 F(\xi, \eta) d\eta d\xi \approx \sum_{k=0}^n \sum_{l=0}^n \omega_k \omega_l F(\xi_k \eta_l), \quad (2.39)$$

où les valeurs de (ξ_k, η_l) et ω_m peuvent être calculées. Le développement qui permet de les obtenir est décrit dans [17]. Les valeurs dont nous aurons besoin sont les suivantes :

Ordre	Points	$\xi, \eta, -\xi, -\eta$	ω
1	1	0.00000	2.00000
2	4	0.57735	1.00000
6	36	0.93247	0.17132
		0.66121	0.36076
		0.23862	0.46791

3 Schémas dynamiques

Le programme de calcul est un programme dynamique. Il permet de calculer l'évolution dans le temps de la déformation d'une membrane soumise à un jeu de forces.

Pour trouver la déformation de la membrane sous une charge qui évolue dans le temps, nous discrétisons celui-ci en un certain nombre d'intervalles sur lesquels nous calculons ces déformations. Nous déterminons donc, en chacun des « nœuds temporels » t , la position, la vitesse et l'accélération $(\mathbf{X}_t, \mathbf{V}_t, \mathbf{A}_t)$ des nœuds à partir de $(\mathbf{X}_{t-1}, \mathbf{V}_{t-1}, \mathbf{A}_{t-1})$ et des forces aux nœuds via un algorithme *step by step*.

3.1 Equation générale

L'approche la plus générale pour trouver la réponse dynamique d'une structure est l'intégration numérique directe des équations dynamiques [18]. Trois types d'énergies sont mises en jeu lorsque l'analyse dynamique d'une structure est pratiquée :

- l'énergie de déformation élastique,
- l'énergie dissipée par l'amortissement,
- l'énergie cinétique.

L'équation différentielle qui régit le mouvement d'une structure amortie soumise à une sollicitation dynamique $\mathbf{F}(t)$ est donnée par :

$$\mathbf{M} \cdot \ddot{\mathbf{X}}(t) + \mathbf{C} \cdot \dot{\mathbf{X}}(t) + \mathbf{K} \cdot \mathbf{X}(t) = \mathbf{F}(t), \quad (2.40)$$

où \mathbf{M} , \mathbf{C} et \mathbf{K} sont les matrices de masse, d'amortissement et de raideur du système discrétisé entier (matrices assemblées), et \mathbf{X} , $\dot{\mathbf{X}}$ et $\ddot{\mathbf{X}}$ sont les positions, vitesses et accélérations de l'ensemble des nœuds.

C'est cette équation dynamique que nous allons résoudre, et pour ce faire nous avons examiné deux possibilités. L'une est un algorithme implicite de Newmark, l'autre est un algorithme explicite. Nous avons mis en œuvre le second pour des raisons que nous expliquons dans la suite de cette section. Son désavantage principal est qu'il est conditionnellement stable, c'est-à-dire qu'il existe un pas de temps critique Δt_{crit} au-delà duquel l'algorithme est instable.

3.2 Schémas explicites et implicites

Les différents algorithmes utilisés pour la résolution de cette équation se classent généralement en deux catégories : les implicites et les explicites. Nous décrivons les deux méthodes en présentant la résolution d'un problème classique : le problème de Cauchy. Le problème de Cauchy consiste à rechercher une fonction $u(x)$ telle que :

$$\dot{u}(x) = f(u(x), x), \quad x \in [a, b], \quad (2.41)$$

$$u(a) = u_0. \quad (2.42)$$

Etant donnée une valeur de $u_0 \in \mathfrak{R}$, nous cherchons la fonction

$$u : x \in \mathfrak{R}^+ \longrightarrow u(x) \in \mathfrak{R}, \quad (2.43)$$

qui satisfait les équations ci-dessus. Il n'est pas toujours aisé de trouver cette solution. C'est pourquoi nous avons recours à des techniques d'analyse numérique pour l'évaluer. La méthode d'Euler permet de tracer point par point une courbe approchée de celle de $u(x)$.

Méthode d'Euler

La méthode d'Euler [19] est une méthode numérique qui consiste à obtenir une courbe approchée d'une fonction qui vérifie une équation différentielle du premier ordre et une condition initiale d'Euler :

$$\frac{\partial u}{\partial x} = f(x, u(x)), \quad (2.44)$$

$$u(x_0) = u_0. \quad (2.45)$$

Le schéma d'Euler est un schéma d'approximation itératif. Il consiste à subdiviser le domaine du temps ou de l'espace sur lequel nous souhaitons approximer la fonction en un certain nombre d'intervalles, et à calculer par itérations successives les valeurs approchées de la solution à leurs bornes.

Nous divisons donc le domaine en n intervalles dont les bornes sont $a = X_0, X_1 = X_0 + h, X_2 = X_1 + h, \dots, X_{n+1} = X_n + h$, et nous prenons le pas d'intégration $h = \frac{X_n - X_0}{n}$. Les valeurs estimées seront d'autant plus proches des valeurs exactes que le pas h est petit.

La méthode d'Euler se base sur la propriété suivante :

$$u(X_i + h) \approx u(X_i) + hu'(X_i) = u(X_i) + hf(X_i), \quad (2.46)$$

grâce à laquelle nous pouvons calculer de proche en proche les n valeurs approchées $u(X_1), u(X_2), \dots, u(X_n)$ à partir de la condition initiale. En écrivant U_i l'approximation de $u(X_i)$, cela donne :

$$u(X_0 + h) \approx u(X_0) + hu'(X_0) \quad \text{soit} \quad u(X_1) \approx U_0 + hf(X_0) = U_1 \quad (2.47)$$

$$u(X_1 + h) \approx u(X_1) + hu'(X_1) \quad \text{soit} \quad u(X_2) \approx U_1 + hf(X_1) = U_2 \quad (2.48)$$

...

$$u(X_{n-1} + h) \approx u(X_{n-1}) + hu'(X_{n-1}) \quad \text{soit} \quad u(X_n) \approx U_{n-1} + hf(X_{n-1}) = U_n \quad (2.49)$$

La figure 2.11 représente la démarche. La courbe constituée des segments approche la courbe exacte.

Les différentes méthodes d'Euler se distinguent par le choix de l'approximation de la fonction f . Nous pouvons classer ces méthodes en deux catégories : explicites et implicites.

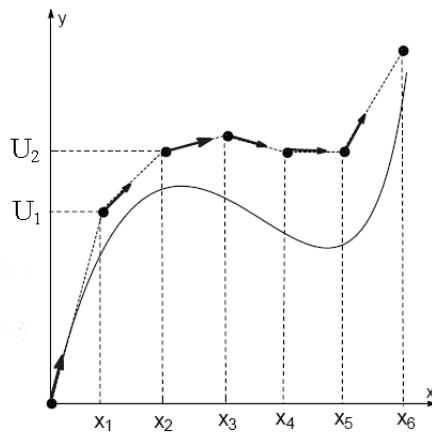


FIG. 2.11 – Illustration de la méthode d'Euler.

Schéma explicite

Dans un schéma explicite, l'évaluation de U_{i+1} se fait grâce à U_i et la dérivée de la fonction $u(x)$ en X_i .

$$U_{i+1} = U_i + f(U_i, X_i)h \quad \text{avec} \quad U(0) = u_0. \quad (2.50)$$

Nous pouvons donc obtenir U_i en partant de u_0 et en itérant i fois (figure 2.12).

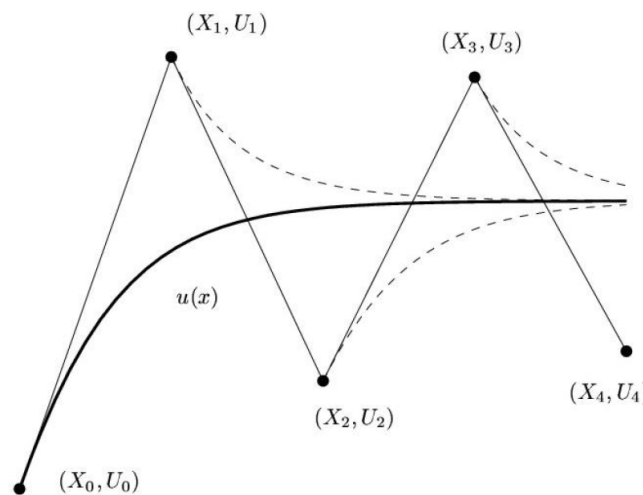


FIG. 2.12 – Illustration de la méthode explicite.

L'implémentation de ce schéma est souvent beaucoup plus aisée que pour un schéma implicite. Malheureusement, ces méthodes ont un domaine de validité pour le pas Δx hors duquel elles sont instables.

En intégrant le problème (2.45) :

$$\int_{X_i}^{X_{i+1}} f(x, u(x)) dx = u(X_{i+1}) - u(X_i), \quad (2.51)$$

et en réécrivant (2.50) :

$$hf(X_i, U_i) \simeq U_{i+1} - U_i, \quad (2.52)$$

nous voyons que nous pouvons approximer l'intégrale par l'aire d'un rectangle [20] (figure 2.14).

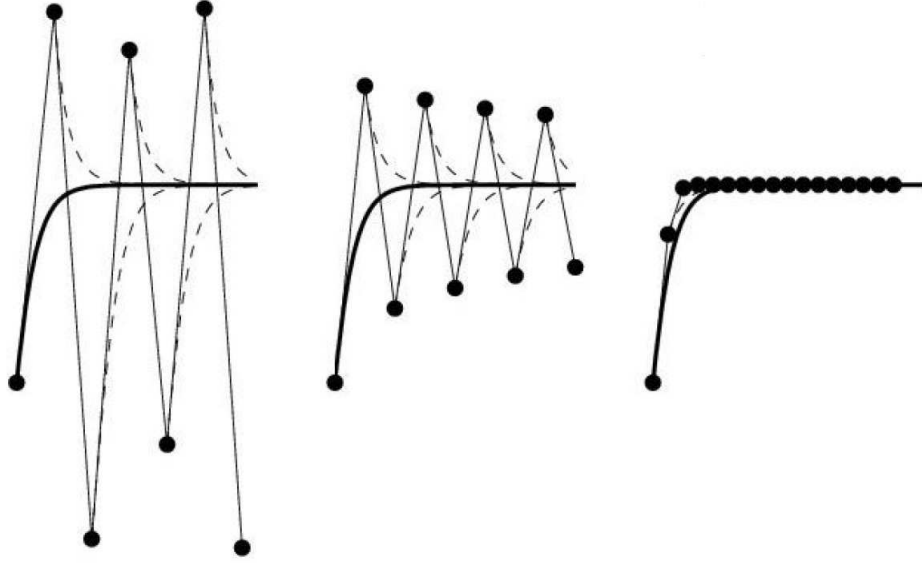


FIG. 2.13 – Stabilité du schéma explicite.

Intéressons-nous à la figure 2.13, qui fournit une illustration de la stabilité d'une méthode explicite et met en évidence l'importance du pas de temps. Il apparaît très clairement que l'instabilité de l'image de gauche est due à un pas trop grand, c'est-à-dire une utilisation de $f(x_0)$ sur un intervalle du domaine trop long. L'approximation ne s'adapte donc pas aux mouvements brusques de $u(x)$. Par contre, la dernière image montre bien la fidélité de l'approximation pour un très petit pas de temps.

Schéma implicite

Dans un schéma implicite, l'évaluation de U_{i+1} se fait grâce à U_i et U_{i+1} :

$$U_{i+1} = U_i + f(U_{i+1}, X_{i+1})h_i \quad \text{avec} \quad U(0) = u_o. \quad (2.53)$$

Comme précédemment, nous pouvons récrire le problème :

$$hf(X_{i+1}, U_{i+1}) \simeq U_{i+1} - U_i, \quad (2.54)$$

et nous pouvons approximer l'intégrale par un rectangle différent du cas explicite (figure 2.14). Ces méthodes demandent la résolution d'un système d'équations linéaires à chaque pas de temps, mais des pas de temps plus grands peuvent être utilisés car la plupart de ces méthodes sont assez stables, voire inconditionnellement stables.

Signalons également une autre méthode : *la méthode Crank-Nicolson* [20].

$$U_{i+1} = U_i + \frac{f(U_i, X_i) + f(U_{i+1}, X_{i+1})}{2} h_i. \quad (2.55)$$

Il s'agit d'une méthode semi-implicite qui permet de conserver l'énergie de façon discrète. L'approximation de l'intégrale se fait par des trapèzes.

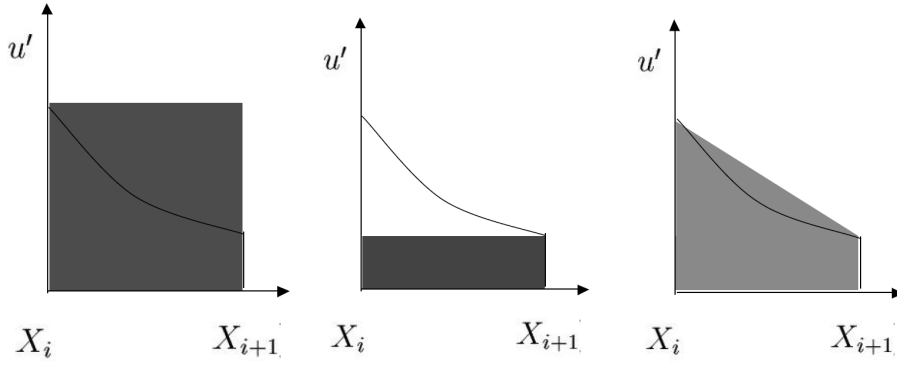


FIG. 2.14 – Evaluation de l'intégrale pour les schémas explicites, implicites et de Crank-Nicolson.

3.3 Algorithme implicite : équations de Newmark

Newmark a présenté une famille de méthodes d'intégration à pas unique pour le calcul de la solution du problème dynamique défini par l'équation du mouvement (2.40) que nous rappelons :

$$\mathbf{M} \cdot \ddot{\mathbf{X}}(t) + \mathbf{C} \cdot \dot{\mathbf{X}}(t) + \mathbf{K} \cdot \mathbf{X}(t) = \mathbf{F}_{\text{ext}}(t), \quad (2.56)$$

ou, en laissant de côté le terme d'amortissement (celui-ci sera abordé dans une autre section), et en considérant que $\mathbf{K} \cdot \mathbf{X}(t) = \mathbf{F}_{\text{int}}(t)$:

$$\mathbf{M} \cdot \ddot{\mathbf{X}}(t) = \mathbf{F}_{\text{ext}}(t) - \mathbf{F}_{\text{int}}(t), \quad (2.57)$$

ce qui correspond bien au résultat de la discrétisation du Principe des Travaux Virtuels (équation 2.15).

Les équations de Newmark nous donnent \mathbf{X}_{t+1} , \mathbf{V}_{t+1} et \mathbf{A}_{t+1} à partir de \mathbf{X}_t , \mathbf{V}_t

et \mathbf{A}_t et ont la forme suivante :

$$\mathbf{X}_{t+1} = \mathbf{X}_t + \Delta t \dot{\mathbf{X}}_t + (1/2 - \beta) \Delta t^2 \ddot{\mathbf{X}}_t + \beta \Delta t^2 \ddot{\mathbf{X}}_{t+1}, \quad (2.58)$$

$$\dot{\mathbf{X}}_{t+1} = \dot{\mathbf{X}}_t + (1 - \gamma) \Delta t \ddot{\mathbf{X}}_t + \gamma \Delta t \ddot{\mathbf{X}}_{t+1}, \quad (2.59)$$

$$\ddot{\mathbf{X}}_t = \mathbf{M}^{-1} \cdot \left(\mathbf{F}^{\text{ext}}(t, \mathbf{X}_t) - \mathbf{F}^{\text{int}}(\mathbf{X}_t) \right), \quad (2.60)$$

$$\ddot{\mathbf{X}}_{t+1} = \mathbf{M}^{-1} \cdot \left(\mathbf{F}^{\text{ext}}(t+1, \mathbf{X}_{t+1}) - \mathbf{F}^{\text{int}}(\mathbf{X}_{t+1}) \right), \quad (2.61)$$

où $\mathbf{F}^{\text{int}}(\mathbf{X}_t)$ et $\mathbf{F}^{\text{ext}}(t, \mathbf{X}_t)$ sont les forces intérieures et extérieures au temps t décrites dans la section précédente, \mathbf{M} est la matrice de masse du système,

$$\left| \begin{array}{ll} \mathbf{X}_t & \text{est le déplacement au temps } t, \\ \mathbf{X}_{t+1} & \text{est le déplacement au temps } t+1, \\ \dot{\mathbf{X}}_t & \text{est la vitesse au temps } t, \\ \ddot{\mathbf{X}}_t & \text{est l'accélération au temps } t, \end{array} \right.$$

et β et γ sont des coefficients qui caractérisent la méthode particulière de Newmark qui est utilisée.

Il s'agit bien d'équations implicites, puisque le calcul de \mathbf{X}_{t+1} et \mathbf{V}_{t+1} nécessite celui de \mathbf{A}_{t+1} , alors que ce dernier dépend de \mathbf{X}_{t+1} .

Puisque nous n'avons finalement pas implémenté de méthode implicite, nous parlerons de cet algorithme sans donner de valeurs particulières pour β et γ . Notons toutefois que la méthode de Newmark est inconditionnellement stable pour :

$$2\beta > \gamma > 1/2. \quad (2.62)$$

Méthode des prédicteurs

Pour résoudre des équations implicites, nous pouvons utiliser une méthode itérative de *prédicteurs-correcteurs*, c'est-à-dire une technique au cours de laquelle nous réalisons une estimation des valeurs recherchées, que nous modifions plusieurs fois par la suite à l'aide d'une formule de correction afin de se rapprocher de la solution exacte.

Le *prédicteur* est donné par les équations explicites équivalentes :

$$\mathbf{X}_{t+1} = \mathbf{X}_t + \Delta t \dot{\mathbf{X}}_t + (1/2 - \beta) \Delta t^2 \ddot{\mathbf{X}}_t, \quad (2.63)$$

$$\dot{\mathbf{X}}_{t+1} = \dot{\mathbf{X}}_t + (1 - \gamma) \Delta t \ddot{\mathbf{X}}_t, \quad (2.64)$$

$$\ddot{\mathbf{X}}_{t+1} = 0. \quad (2.65)$$

Il faut ensuite corriger ces valeurs par un *correcteur* ($\Delta \mathbf{x}, \Delta \mathbf{v}, \Delta \mathbf{a}$) à chaque itération, qui sera apporté jusqu'à l'itération $k+1$:

$$\mathbf{X}_{t+1}^{k+1} = \mathbf{X}_{t+1}^k + \Delta \mathbf{x} = \mathbf{X}_{t+1}^k + \beta \Delta \mathbf{a} \Delta t^2, \quad (2.66)$$

$$\dot{\mathbf{X}}_{t+1}^{k+1} = \dot{\mathbf{X}}_{t+1}^k + \Delta \mathbf{v} = \dot{\mathbf{X}}_{t+1}^k + \gamma \Delta \mathbf{a} \Delta t, \quad (2.67)$$

$$\ddot{\mathbf{X}}_{t+1}^{k+1} = \ddot{\mathbf{X}}_{t+1}^k + \Delta \mathbf{a}. \quad (2.68)$$

Par les deux premières équations nous pouvons déterminer les correcteurs $\Delta \mathbf{a}$ et $\Delta \mathbf{v}$ en fonction de $\Delta \mathbf{x}$:

$$\Delta \mathbf{a} = \frac{\Delta \mathbf{x}}{\beta \Delta t^2}, \quad \Delta \mathbf{v} = \frac{\gamma \Delta \mathbf{x}}{\beta \Delta t}. \quad (2.69)$$

Pour déterminer $\Delta \mathbf{x}$, nous introduisons les formules de correction dans l'équation (2.61) :

$$\mathbf{M} \cdot (\ddot{\mathbf{X}}_{t+1}^k + \Delta \mathbf{a}) + \mathbf{F}^{\text{int}}(\mathbf{X}_{t+1}^k + \Delta \mathbf{x}) = \mathbf{F}^{\text{ext}}(t+1, \mathbf{X}_{t+1}^k), \quad (2.70)$$

$$\begin{aligned} \mathbf{M} \cdot (\ddot{\mathbf{X}}_{t+1}^k + \Delta \mathbf{a}) + \mathbf{F}^{\text{int}}(\mathbf{X}_{t+1}^k) + \frac{\partial \mathbf{F}^{\text{int}}}{\partial \mathbf{X}} \Big|_{\mathbf{X}_{t+1}^k} \cdot \Delta \mathbf{x} \\ \approx \mathbf{F}^{\text{ext}}(t+1, \mathbf{X}_{t+1}^k) + \frac{\partial \mathbf{F}^{\text{ext}}}{\partial \mathbf{X}} \Big|_{\mathbf{X}_{t+1}^k} \cdot \Delta \mathbf{x}. \end{aligned} \quad (2.71)$$

En exprimant les termes de raideur par :

$$\mathbf{K}_{NM}^{\text{int}} = \frac{\partial \mathbf{F}_N^{\text{int}}}{\partial \mathbf{X}_M} \Big|_{(\mathbf{X}_{t+1}^k)}, \quad (2.72)$$

$$\mathbf{K}_{NM}^{\text{ext}} = \frac{\partial \mathbf{F}_N^{\text{ext}}}{\partial \mathbf{X}_M} \Big|_{(\mathbf{X}_{t+1}^k)}, \quad (2.73)$$

nous obtenons l'équation :

$$\mathbf{M} \cdot \Delta \mathbf{a} + (\mathbf{K}_{t+1}^{\text{int}} - \mathbf{K}_{t+1}^{\text{ext}}) \cdot \Delta \mathbf{x} = \mathbf{F}_{t+1}^{\text{ext}} - \mathbf{M} \cdot \ddot{\mathbf{X}}_{t+1}^k - \mathbf{F}^{\text{int}}(\mathbf{X}_{t+1}^k). \quad (2.74)$$

En définissant l'opérateur tangent \mathbf{K}^{T} par :

$$\mathbf{K}^{\text{T}} = \mathbf{K}^{\text{int}} - \mathbf{K}^{\text{ext}}, \quad (2.75)$$

et en insérant (2.69), nous obtenons :

$$\left(\frac{1}{\beta \Delta t^2} \mathbf{M} + (\mathbf{K}^{\text{T}})_{t+1}^k \right) \cdot \Delta \mathbf{x} = \mathbf{F}_{t+1}^{\text{ext}} - \mathbf{M} \cdot \ddot{\mathbf{X}}_{t+1}^k - \mathbf{F}^{\text{int}}(\mathbf{X}_{t+1}^k). \quad (2.76)$$

Le problème se réduit alors à :

$$(\mathbf{K}_{\text{eff}}^{\text{T}})_{t+1}^k \cdot \Delta \mathbf{x} = \mathbf{\Gamma}_{t+1}, \quad (2.77)$$

avec
$$\begin{cases} \mathbf{K}_{\text{eff}}^{\text{T}} = \left(\frac{1}{\beta \Delta t^2} \mathbf{M} + \mathbf{K}^{\text{T}} \right), \\ \mathbf{\Gamma} = \mathbf{F}^{\text{ext}} - \mathbf{M} \cdot \ddot{\mathbf{X}} - \mathbf{F}^{\text{int}}. \end{cases}$$

Nous pouvons donc trouver le correcteur $\Delta \mathbf{x}$ et ainsi déterminer l'ensemble des correcteurs du modèle prédicteur-correcteur. Il nous reste malgré tout encore à calculer la matrice \mathbf{K}^{T} relative à un élément de la membrane. Il faut pour se faire trouver les dérivées de \mathbf{F}^{int} et \mathbf{F}^{ext} par rapport à \mathbf{X} :

$$\mathbf{K}_{\mathbf{e}}^{\text{int}} = \frac{\partial \mathbf{F}_{\mathbf{e}}^{\text{int}}}{\partial \mathbf{X}_{\mathbf{e}}} = \int_{-1}^1 \int_{-1}^1 H \frac{\partial \mathbf{B}_{\alpha\beta}^T S^{\beta\alpha}}{\partial \mathbf{X}_{\mathbf{e}}} \parallel \mathbf{G}_1 \wedge \mathbf{G}_2 \parallel d\theta^1 d\theta^2, \quad (2.78)$$

où le terme $\mathbf{k}_e^{\text{int}} = \frac{\partial \mathbf{B}_{\alpha\beta}^T S^{\beta\alpha}}{\partial \mathbf{X}_e}$ dépend d'un tenseur d'ordre quatre, $\frac{\partial \mathbf{S}}{\partial \mathbf{C}}$.

$$\mathbf{K}_e^{\text{ext}} = \frac{\partial \mathbf{F}_e^{\text{ext}}}{\partial \mathbf{X}_e} = \int_{-1}^1 \int_{-1}^1 p \mathbf{N}^T \cdot \frac{\partial (\mathbf{g}_1 \wedge \mathbf{g}_2)}{\partial \mathbf{X}_e} d\theta^1 d\theta^2. \quad (2.79)$$

Les intégrales peuvent s'obtenir par une intégration numérique de Gauss-Legendre.

3.4 Schéma explicite

Avec le schéma proposé ci-dessus, une itération serait coûteuse en temps car le calcul du correcteur $\Delta \mathbf{x}$ fait intervenir des tenseurs d'ordre 4, et nécessite de les inverser. Par ailleurs, la mise au point d'un tel code serait longue et nous manquerions de temps pour l'implémenter. Ajoutons enfin que, puisque nous résolvons une équation de dynamique des structures, notre but n'est pas de tendre le plus rapidement possible vers une solution, mais plutôt de représenter régulièrement les déformations de la membrane au cours du temps. Nous n'aurons donc pas besoin de pas de temps trop grands. Pour toutes ces raisons, nous avons préféré utiliser un algorithme explicite.

Pour calculer les positions des nœuds au temps $t + 1$, nous utilisons une méthode d'Euler définie par :

$$\mathbf{X}_{t+1} = \mathbf{X}_t + \Delta t \dot{\mathbf{X}}_t + \frac{\Delta t^2}{2} \ddot{\mathbf{X}}_t, \quad (2.80)$$

où les vitesses sont obtenues par différences finies et les accélérations sont déduites de (2.57) :

$$\dot{\mathbf{X}}_t = \frac{\mathbf{X}_t - \mathbf{X}_{t-1}}{\Delta t}, \quad (2.81)$$

$$\ddot{\mathbf{X}}_t = \mathbf{M}^{-1} \cdot (\mathbf{F}_{\text{ext}}^t - \mathbf{F}_{\text{int}}^t). \quad (2.82)$$

Pas de temps limite

Pour un problème à une dimension, l'algorithme sera stable si une onde ne peut pas traverser complètement un élément au cours d'une itération. Cela revient à dire que, au niveau des ordres de grandeurs, la longueur des éléments doit être supérieure au produit de la vitesse de l'onde par le pas de temps :

$$dt_{\text{crit}} \simeq C^{\text{ste}} \frac{l_{\text{elem}}}{c}, \quad (2.83)$$

où c est la vitesse d'une onde de traction ou compression.

Dans notre cas, la théorie renseigne que la constante vaut $\frac{1}{3}$ environ :

$$dt_{\text{crit}} \simeq \frac{1}{3} \frac{l_{\text{elem}}}{c}, \quad (2.84)$$

où, par sécurité, la longueur l_{elem} est prise égale au rayon r du cercle inscrit au « plus petit » élément.

La vitesse c est égale à la fréquence propre f_{elem} de l'élément multipliée par sa longueur (ou plus exactement le rayon du cercle inscrit), ce qui permet d'écrire le pas de temps critique :

$$dt_{crit} \simeq \frac{r}{3r\sqrt{\frac{k}{m}}} = \frac{r\sqrt{rH\rho}}{3r\sqrt{\frac{HC_1}{r}}} = \frac{r}{3}\sqrt{\frac{\rho}{C_1}}, \quad (2.85)$$

en considérant que $f_{elem} = \sqrt{\frac{k}{m}}$.

Pour vérifier cette expression, nous avons effectué une série de tests sur chaque paramètre. Le résultat obtenu dans le pire des cas est proche de la théorie :

$$dt_{crit} \simeq 0.37r\sqrt{\frac{\rho}{C_1}}. \quad (2.86)$$

La figure 2.15 montre la déformation d'une membrane cylindrique avec deux pas de temps différents : l'un est inférieur au pas de temps critique, l'autre est trop élevé, ce qui rend l'algorithme instable.

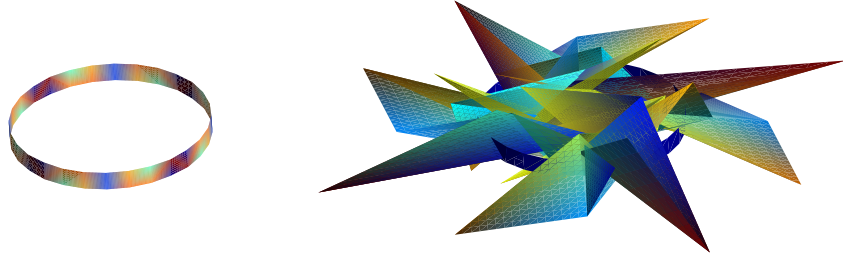


FIG. 2.15 – Illustration de la limite du pas de temps : sur la deuxième image, le pas de temps utilisé est supérieur à dt_{crit} .

3.5 Amortissement

La résolution de l'équation de la dynamique des structure sans terme d'amortissement mène à des résultats oscillant indéfiniment. Si nous voulons modéliser les dissipations survenant au sein du matériau et ainsi mieux approcher la réalité, nous avons besoin d'exprimer les pertes dans notre schéma de résolution. Pour ce faire, nous utilisons le terme d'amortissement dans l'équation (2.40) de la dynamique et nous l'introduisons dans notre algorithme :

$$\ddot{\mathbf{X}}_{t-1} = \mathbf{M}^{-1} \cdot (\mathbf{F}_{ext}^t - \mathbf{F}_{int}^t - \mathbf{C}_{amort} \cdot \dot{\mathbf{X}}_{t-1}). \quad (2.87)$$

Puisque nous cherchons uniquement à provoquer une perte d'énergie, nous n'avons pas besoin de calculer une matrice d'amortissement complexe. Nous utiliserons donc une matrice \mathbf{C}_{amort} [kg/s] diagonale, correspondant à un amortissement visqueux [21].

La figure 2.16 montre l'évolution du rayon d'une membrane cylindrique de rayon 10 cm et d'épaisseur 5 mm (longueur infinie) soumise brutalement à une différence de pression intérieure/extérieure de 100 kg/m^2 avec les constantes de raideur $C_1 = 10^5\text{ Pa}$ et $C_2 = 0\text{ Pa}$, pour plusieurs valeurs de C_{amort} . Les valeurs d'amortissements y sont renseignées en kg/s .

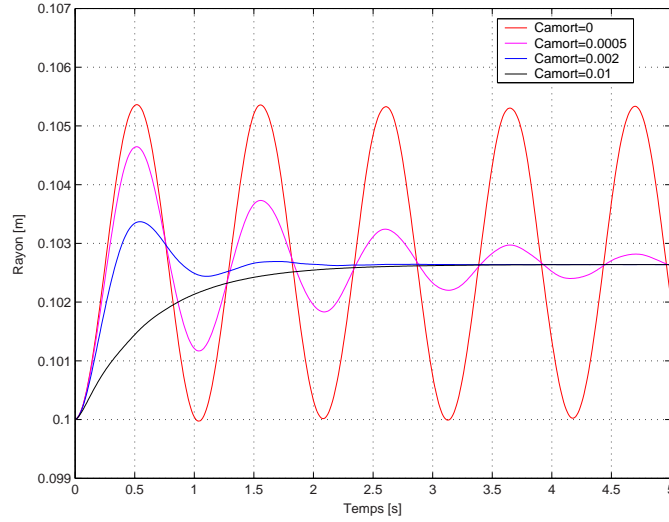


FIG. 2.16 – Evolution du rayon d'un cylindre infini soumis à une pression constante, sous différents amortissements $[\text{kg/s}]$.

3.6 Conditions aux limites

Des conditions aux frontières peuvent être imposées à la membrane. Nous avons étudié la possibilité de bloquer certains nœuds le long d'un segment ou en un point précis. Le blocage peut se faire dans l'une des trois directions $(\vec{e}_x, \vec{e}_y, \vec{e}_z)$ ou dans plusieurs de celles-ci.

L'imposition de ces conditions se fait en annulant les composantes correspondantes de la vitesse et de l'accélération avant le calcul d'une nouvelle position.

Les conditions initiales par défaut sont un déplacement initial et une vitesse initiale nuls. Toutefois, nous pouvons déplacer arbitrairement un ou plusieurs nœuds avant ou pendant l'exécution. Nous devrions alors observer un retour de la membrane vers sa position de référence. De la même façon, il est possible d'imposer une vitesse initiale non nulle à certains nœuds.

La combinaison d'un déplacement initial de nœuds et de conditions aux frontières empêchant leur mouvement donne lieu à une traction équivalente à une précontrainte en génie civil.

Chapitre 3

Implémentation

Dans ce chapitre, nous décrivons les moyens que nous avons mis en œuvre pour implémenter les différents concepts présentés auparavant et mettre ainsi sur pied un code de calcul de dynamique pour les membranes. Nous avons choisi de travailler avec le logiciel *Matlab*¹ pour l'implémentation, et ceci pour plusieurs raisons :

1. le programme « fluide » que nous utilisons pour l'interaction fluide-structure possède un module facilitant le couplage avec un code Matlab,
2. le langage utilisé par Matlab nous est familier, car nous l'avons utilisé à plusieurs reprises par le passé,
3. ce programme est idéal pour travailler en langage matriciel, ce qui s'est avéré très utile tout au long de l'écriture du code.

1 Description du logiciel *Gmsh*

Nous utilisons le logiciel *Gmsh*, développé par Christophe Geuzaine et Jean-François Remacle, pour réaliser les maillages bidimensionnels et tridimensionnels sur lesquels nous travaillons. Nous l'utilisons également pour visualiser nos résultats.

1.1 Les fonctions

Le logiciel possède différents modules. Trois de ceux-ci nous sont particulièrement utiles : les modules de la géométrie, du maillage et du post-processing.

Géométrie : *Gmsh* nous permet de produire une grande variété de formes allant du simple cylindre au parachute détaillé. La construction d'une géométrie permet l'emploi de paramètres sur les longueurs d'arêtes, le nombre de mailles, la

¹Une description du logiciel est disponible sur Internet via l'adresse <http://www.mathworks.com/products/matlab/description1.html>

longueur des mailles, ... ce qui rend facile l'adaptation d'un maillage à un nouveau problème. La plus grosse partie de la géométrie est faite via une interface, tandis que certains problèmes sont traités directement en éditant le fichier de commandes de création de la géométrie.

Tout élément, surface, ligne ou point peut être considéré par le programme comme étant « physique », c'est-à-dire qu'une information supplémentaire, appelé *tag*, correspondant à une propriété physique sera attribuée à l'entité, et transmise dans le fichier contenant le maillage fini. Cela permet de différencier physiquement les différents points, arêtes, surfaces et volumes s'ils ont des propriétés particulières.

Mailleur : Le logiciel permet de produire un maillage de triangles sur les surfaces définies par la géométrie. Le nombre et la taille des mailles sont définis via des paramètres attachés aux frontières de ces surfaces. Le maillage peut se faire suivant certaines contraintes de régularité des mailles (lignes et surfaces dites *transfinies*) avec un nombre de nœuds imposé sur une arête, ou bien par la définition d'une longueur approximative de côté de mailles.

Les mailles peuvent ensuite être recombinaées de manière à obtenir des quadrangles. Dans le cas d'un maillage régulier, la recombinaison peut être totale et donner ainsi lieu à un maillage exclusivement quadrangulaire.

Le mailleur permet de mailler dans une, deux ou trois dimensions, en formant successivement des éléments linéaires, surfaciques et volumiques. Les éléments surfaciques sont ceux expliqués plus haut. Les éléments volumiques sont des tétraèdres. Le programme que nous utiliserons pour calculer les mouvements et pressions du fluide utilise des maillages de tétraèdres.

Post-processing : Les résultats peuvent être visualisés via le post-processeur de *Gmsh*. La syntaxe utilisée dans le fichier d'entrée est assez simple, de sorte qu'il est possible de traduire nos résultats de déformations de membranes en un fichier lisible par le logiciel. C'est l'objet de la fonction *exportNodalField2Gmsh.m*.

Le code *Sower* permet par ailleurs de convertir des résultats provenant du programme de calcul des fluides en fichiers lisibles par *Gmsh*.

Une fois les résultats affichés, le programme fournit plusieurs outils permettant de les visualiser sous différentes formes.

1.2 Le maillage produit

Le maillage produit est traduit en termes de nœuds, lignes, surfaces et volumes dans un fichier *.msh*. La forme du fichier est décrit ci-dessous :

```

NOD
Nombre de Nœuds
numéro du nœud1   Xnœud1   Ynœud1   Znœud1
...               ...       ...       ...
ENDNOD
ELM
Nbr elem
num elem          type elem  physical  reg elem  nbr nœuds  num nœuds
...               ...       ...       ...       ...       ...
ENDELM

```

où :

Nbr elem = nombre d'éléments (lignes, arêtes, triangles et/ou quadrangles),

num elem = numéro de l'élément,

type elem = type d'élément (1 = arête, 2 = triangle, 3 = quadrangle),

physical = tag attribué à l'élément,

reg elem = numéro de l'entité élémentaire auquel appartient l'élément,

nbr nœuds = nombre de nœuds de l'élément,

num nœuds = les numéros des nœuds de l'élément.

2 Description du programme

Nous présentons ici toutes les pièces du puzzle permettant de passer d'un maillage produit par le logiciel *Gmsh*, et un jeu de forces s'exerçant sur ce maillage, à un résultat sous la forme de déplacements au cours du temps.

2.1 Schéma général

Le code est articulé autour du programme principal *membrane.m*, qui fait appel à chaque fragment de programme nécessaire. La figure 3.1 en montre le fonctionnement général.

Dans un premier temps, *membrane.m* fait appel à *load_gmsh_2D.m*, qui charge le maillage dans une variable de type structure de données, que nous appelons *mesh*. Il réalise ensuite les quatre opérations suivantes :

- calcul du rapport des aires de chaque élément à l'élément parent avec *detJacobiansInit.m* et stockage de cette information dans *mesh*,
- calcul de la matrice de masse globale du système avec *matr_masse.m* et stockage de l'information dans *mesh*,
- initialisation des variables de déplacements, vitesses et accélérations des nœuds en fonction des conditions initiales par *initial.m*,

- conversion des données relatives aux forces ou aux pressions aux nœuds en une variable spécifique par *readP.m*.

Le programme lance ensuite le processus itératif. Au cours d'une itération, les opérations suivantes sont effectuées :

- écriture (éventuelle) d'un fichier de type *.pos* contenant les déplacements actuels et les forces internes dans un format lisible par *Gmsh* avec *writeFile.m*,
- calcul des forces internes et externes par *forces.m*,
- calcul des déplacements, vitesses et accélérations et application des conditions aux frontières par *frontieres.m*.

Si les normes des vitesses et accélérations des nœuds sont toutes proches de 0, c'est-à-dire dans un intervalle $[0 - \epsilon, 0 + \epsilon]$ choisi par l'utilisateur au moment de l'exécution, le processus itératif est arrêté.

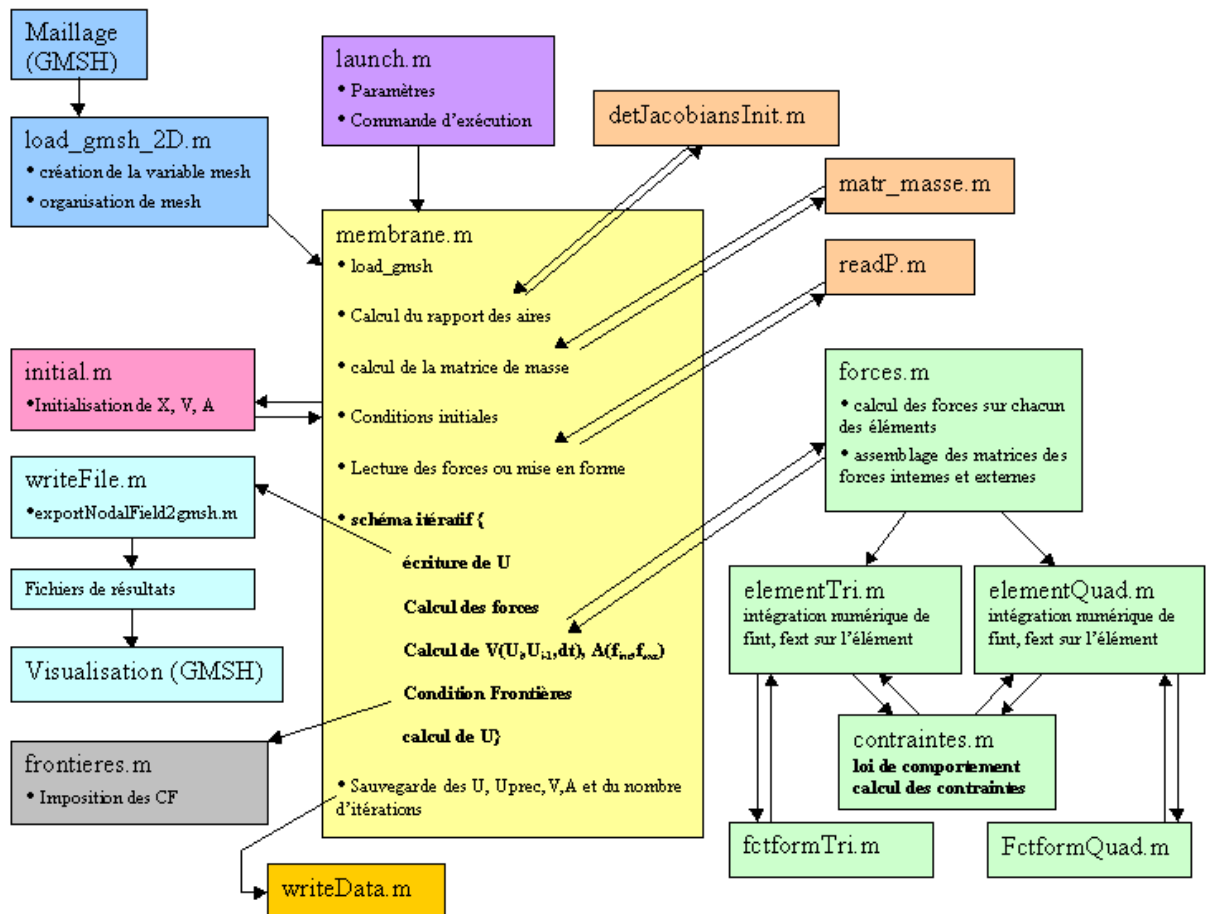


FIG. 3.1 – Fonctionnement général du programme.

Pour terminer, le programme écrit les données de déplacements, vitesses et accélérations des deux derniers pas de temps dans un fichier texte avec *writeData.m*.

2.2 Entrées et sorties du programme

Entrées

Le programme *membrane* reçoit un ensemble de paramètres que nous répartissons en trois catégories : les données d'entrée, les paramètres du calcul et les données relatives aux sorties. Leur détail est le suivant :

Données d'entrée :

- le chemin d'accès et le nom du fichier *.msh* contenant le maillage,
- le chemin d'accès et le nom du fichier contenant les forces, ou une valeur qui doit être prise en compte comme une pression appliquée de manière uniforme sur toute la membrane,
- une variable indiquant s'il faut commencer un nouveau calcul, ou s'il faut initialiser les déplacements, vitesses et accélérations en fonction du dernier fichier de sortie produit.

Paramètres de calcul :

- le pas de temps,
- le nombre de pas de temps,
- le critère d'arrêt ϵ au cas où la solution n'évolue plus,
- l'épaisseur de la membrane,
- sa masse volumique,
- ses coefficients de Mooney-Rivlin,
- l'amortissement utilisé.

Données relatives aux sorties :

- le chemin d'accès et le nom du fichier *.txt* sur lequel seront écrits les déplacements, vitesses et accélérations en fin d'exécution,
- le chemin d'accès et le nom des fichiers *.pos*,
- la fréquence de production de ces fichiers.

Le fichier *launch.m* contient tous les paramètres et lance le programme. Il permet d'utiliser le code de manière simple.

Des entrées supplémentaires apparaîtront lorsque nous parlerons du couplage avec le programme de calcul des fluides.

Sorties

Les résultats de l'exécution de *membrane.m* sont doubles :

- d'une part, le fichier contenant les déplacements, vitesses et accélérations produit en fin d'exécution,
- d'autre part l'ensemble des fichiers *.pos* produits durant le processus itératif.

Nous parlerons d'une sortie supplémentaire dans la partie traitant du couplage fluide-structure.

2.3 Organisation du maillage

C'est le programme *load_gmsh_2D.m* qui va transformer le maillage généré par *Gmsh* (par exemple le fichier *cyindre.msh*) et qui va organiser les données afin qu'elles soient utilisables par le programme « structure ». Il va créer une structure de données que nous appellerons *mesh* en parcourant le fichier *.msh*, et la renverra au programme principal. Cette structure, détaillée dans les tableaux 3.1 et 3.2, sera passée en argument lors de l'exécution de la plupart des sous-programmes.

mesh.nbNod	Nombre de nœuds du maillage,
mesh.nbElm	Nombre d'éléments (lignes, arêtes, triangles et quadrangles) du maillage,
mesh.nbBeams	Nombre d'arêtes,
mesh.nbTriangles	Nombre de triangles,
mesh.nbQuads	Nombre de quadrangles.

TAB. 3.1 – Variables de la structure de données *mesh* contenant le nombre d'éléments de chaque type.

Par après le programme va ajouter à *mesh* certaines variables qu'il vient de calculer ou qui ont été introduites par l'utilisateur ; notamment *mesh.Masse*, la matrice de masse du système ; *mesh.H*, l'épaisseur de la membrane ; *mesh.rho*, sa masse volumique, ... Par contre, le programme ne modifie pas les coordonnées renseignées dans *mesh.POS* lorsqu'un nouvel état de déformation a été calculé. La variable contient donc toujours la configuration de référence de la membrane, c'est-à-dire la configuration telle qu'il n'existe aucune contrainte interne.

mesh.POS

Positions des nœuds du maillage, ordonnées de la manière suivante :

	coordonnées		
	X	Y	Z
1 ^{er} nœud	.	.	.
2 ^{ème} nœud	.	.	.
⋮	⋮	⋮	⋮
<i>n</i> ^{ème} nœud	.	.	.

mesh.POINTS

Description de tous les points à tag du système, sous la forme suivante :

	N° nœud	tag
1 ^{er} point	.	.
2 ^{ème} point	.	.
⋮	⋮	⋮
<i>n</i> ^{ème} point	.	.

mesh.BEAMS

Description de toutes les arêtes du système, sous la forme suivante :

	N° nœud 1	N° nœud 2	tag
1 ^{ère} arête	.	.	.
2 ^{ème} arête	.	.	.
⋮	⋮	⋮	⋮
<i>n</i> ^{ème} arête	.	.	.

mesh.TRIANGLES

Description de tous les triangles du système, sous la forme suivante :

	N° 1	N° 2	N° 3	tag
1 ^{er} triangle
2 ^{ème} triangle
⋮	⋮	⋮	⋮	⋮
<i>n</i> ^{ème} triangle

mesh.QUADS

Description de tous les quadrangles du système, sous la forme suivante :

	N° 1	N° 2	N° 3	N° 4	tag
1 ^{er} quadrangle
2 ^{ème} quadrangle
⋮	⋮	⋮	⋮	⋮	⋮
<i>n</i> ^{ème} quadrangle

TAB. 3.2 – Description des éléments dans la structure de données *mesh*.

2.4 Mixité triangles-quadrangles

Dans la section consacrée à la méthode des éléments finis, nous expliquions qu'il était indispensable que notre code admette des maillages triangulaires afin de ne pas complexifier exagérément le couplage fluide-structure. Nous n'avons toutefois pas résisté à la tentation d'ajouter quelques briques à notre programme afin qu'il puisse également traiter un maillage quadrangulaire.

Comme expliqué plus haut, cela implique l'introduction de nouvelles fonctions de forme, ainsi qu'une méthode d'intégration différente. Afin de ne pas créer deux programmes différents, ce qui serait très inconfortable et peu efficace, nous avons intégré le traitement des quadrangles dans le programme initial. Finalement, nous avons poussé l'intégration jusqu'à pouvoir prendre en charge un maillage mixte triangles-quadrangles.

Lorsque nous recherchons la matrice de masse du système complet, ou que nous calculons les forces internes ou externes, nous parcourons les éléments en les assemblant systématiquement dans la matrice globale associée, selon l'indice des nœuds affectés à l'élément. Cette opération peut être faite successivement pour les triangles et les quadrangles.

Au niveau de l'implémentation, cette mixité explique la présence simultanée des deux variables *mesh.TRIANGLES* et *mesh.QUADRANGLES* dans la structure de données *mesh*. Cela implique aussi deux traitements séparés pour les deux types d'éléments, malgré un procédé d'assemblage similaire. Ainsi, le programme *forces.m*, par exemple, va d'abord parcourir tous les triangles et faire le calcul des forces sur chacun d'entre eux via le programme *elementTri.m*, qui utilise la technique d'intégration de Hammer (section I.2.2.5) et les fonctions de forme linéaires (section I.2.1.3) avec *fctformTri.m*, avant d'assembler chaque nouveau résultat dans la matrice globale des forces. Il va ensuite faire de même avec les quadrangles via *elementQuad.m*, qui utilise la technique d'intégration de Gauss-Legendre (section I.2.2.5) et les fonctions de forme bilinéaires (section I.2.1.3) avec *fctformQuad.m*.

Les deux programmes *elementTri.m* et *elementQuad.m* font appel au code de *contraintes.m*, qui contient toutes les règles de calcul de la cinématique développées dans le premier chapitre, ainsi que la loi de comportement de Mooney-Rivlin et le calcul du tenseur des contraintes de Piola-Kirchhoff. Ces calculs sont communs aux deux types d'éléments.

2.5 Imposition des conditions aux frontières

Pour imposer les conditions aux frontières décrites dans la section I.2.3.6, nous devons pouvoir modifier la vitesse et l'accélération des *bons nœuds* au *bon moment*. C'est le programme *frontieres.m* qui modifie les matrices de vitesses et d'accélération des nœuds.

Le moment auquel doivent intervenir les conditions aux frontières dépend de l'en-

chaînement de l'algorithme. Au niveau de l'implémentation, il s'agit simplement de trouver la ligne de code où doit être inséré l'appel à *frontieres.m*.

Déterminer quels sont les nœuds dont les vitesses et les accélérations doivent être modifiées, et dans quelles directions, est un peu plus complexe. Cela nécessite des interventions à trois niveaux :

- Le choix d'imposer une condition aux frontières se fait dès le **dessin de l'objet**, c'est-à-dire dans notre cas, en utilisant *Gmsh*. Si nous voulons imposer une condition sur une arête, nous lui donnons un tag. Celui-ci dépendra du code que nous appliquons pour les conditions aux frontières. Le tableau 3.3 donne celui que nous employons. *Gmsh* va alors répercuter les tags sur les arêtes, lors du maillage, puis de l'écriture du fichier *.msh*.

tag	composante(s) annulée(s)		
	X	Y	Z
100	•		
010		•	
001			•
011		•	•
110	•	•	
101	•		•
111	•	•	•

TAB. 3.3 – Tags utilisés pour indiquer l'accrochage d'un point.

- Lorsque ***load_gmsh_2D.m*** lit le maillage et construit la structure *mesh*, il y insère les tags. Dans *mesh.BEAMS*, la troisième colonne leur est réservée, de même que la deuxième colonne de *mesh.POINTS* (cfr. tableau 3.2).
- Dans le code de ***frontieres.m***, une boucle parcourt tous les points et toutes les arêtes et, en fonction des tags qu'elle trouve, impose une valeur aux vitesses et accélérations des nœuds correspondants.

2.6 Imposition de conditions initiales

Les conditions initiales sont imposées lors de l'exécution de *initial.m*. Deux cas peuvent se présenter, selon la valeur passée en paramètre *init* lors de l'exécution de *membrane.m*.

Premier cas : *init* est nul

Dans ce cas, les variables de déplacements, vitesses et accélérations sont initialisées à zéro. La variable *X* (configuration actuelle) est initialisée aux valeurs contenues dans *mesh.POS*. Il est toutefois possible de donner un déplacement initial à des nœuds dont nous connaissons les indices, en ajoutant une courte ligne de code qui modifie *X*.

Faisons une petite clarification à propos des différentes configurations de la membrane. Comme expliqué plus haut, le maillage est importé sous la forme d'une structure de donnée par notre code (*mesh*). C'est la configuration décrite par cette structure qui sera la configuration de référence tout au long de l'exécution du programme. La variable contenant la position des nœuds est *mesh.POS*. Elle ne sera jamais modifiée. Lorsque le code s'exécute, une variable *X* contiendra les positions de chaque nœud dans la configuration déformée. Les forces internes seront toujours calculées en comparant la situation déformée *X* à la situation de référence *mesh.POS*. Le terme des forces externes ne dépendra quant à lui que de la position *X* (équation (2.14)).

Nous pouvons donc « tirer » sur un nœud en ajoutant tout simplement un déplacement à l'entrée correspondante de *X*. Nous nous attendrons alors à voir des forces de rappel s'exercer autour des nœuds déplacés, et à observer la membrane revenir en oscillant (ou pas) autour de sa position de référence.

Second cas : *init* est non-nul

Dans ce cas, le programme *initial.m* va ouvrir le fichier de sauvegarde des derniers résultats et initialisera les déplacements, vitesses et accélérations aux valeurs contenues dans ce fichier. Cette option permet de poursuivre un calcul sans perdre de données.

Chapitre 4

Vérifications et applications du code de calcul

Avant d'utiliser notre programme dans un procédé de couplage, il convient de le tester en comparant certains résultats à la théorie ou à des résultats expérimentaux. Cela permet non seulement de se rassurer quant à la validité du code, mais aussi, durant les premiers essais, de mettre en évidence les éventuelles erreurs.

Nous avons vérifié les résultats de notre programme en l'exécutant sur deux géométries de membranes suffisamment simples pour que nous connaissions certaines de leurs caractéristiques : le cylindre infini et la sphère. Dans les deux cas, la théorie nous livre une relation entre la pression intérieure et la déformation finale (lorsque les oscillations ont disparu).

La visualisation du comportement dynamique des membranes, proche du comportement attendu, dans les nombreuses géométries que nous avons essayées, nous donne également confiance quant à la validité du programme. Par ailleurs, un test de vibration d'une membrane simple est décrit dans la première section.

Au niveau de la stabilité, il est nécessaire de respecter le critère du pas de temps limite. Une fois cette condition remplie, le programme est relativement robuste. Les mouvements amples et brutaux ne le déstabilisent pas.

Nous précisons que toutes les déformations proposées dans cette section ont été calculées via notre programme. Les déformations visibles sont parfois très importantes mais elles représentent toujours les déformations *réelles*, sans facteur d'exagération.

1 Membrane plane

C'est en observant les résultats obtenus sur une simple membrane rectangulaire que nous sommes parvenus à comprendre le mieux les jeux de forces internes et externes qui s'exerçaient sur les éléments. De nombreux problèmes ont été résolus en faisant

tourner notre code sur une membrane carrée formée de deux éléments triangulaires et sur laquelle une traction était appliquée (figure 4.1).

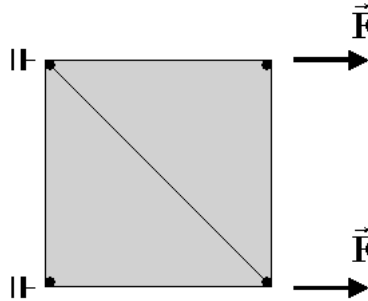


FIG. 4.1 – Traction exercée sur un carré décomposé en deux éléments et fixé sur le côté gauche.

Par la suite, nous avons fait tourner le programme sur un maillage beaucoup plus raffiné de ce même carré et nous avons pu faire quelques observations.

1.1 Membrane au repos

La figure 4.2 montre le maillage utilisé ainsi que l'état déformé après stabilisation.

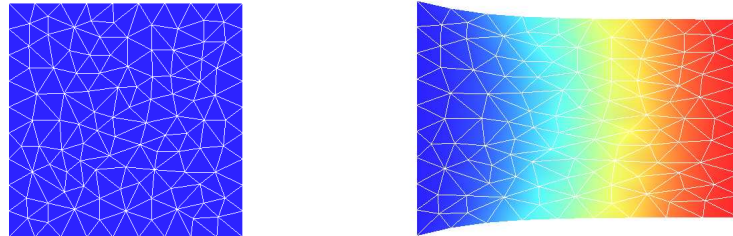


FIG. 4.2 – Résultat d'une traction sur un simple carré attaché par la gauche.

Nous observons que les bords de la membrane décrivent une courbe. Nous pouvions nous attendre à ce résultat vu l'introduction de l'incompressibilité dans notre modèle et vu les efforts perpendiculaires aux tractions principales. En fait, nous pouvons mieux mettre en évidence ce raccourcissement latéral si nous ne fixons pas complètement le bord gauche de la membrane. Nous observons sur la figure 4.3 le changement de forme obtenu dans une configuration où les nœuds du bord gauche ne sont fixés que latéralement.



FIG. 4.3 – Résultat d’une traction sur un simple carré libre verticalement.

Nous pouvons vérifier l’**incompressibilité** de la membrane en regardant les longueurs des côtés du rectangle obtenu. En effet, si L_1 , L_2 , H et V sont les longueurs initiales des côtés de la membrane, sont épaisseur et son volume, et que L'_1 , L'_2 et H' sont les longueurs et épaisseurs après déformation, nous devrions avoir :

$$V = L_1 L_2 H = L'_1 L'_2 H', \quad (4.1)$$

et donc,

$$\lambda_1 = \frac{L'_1}{L_1} = \frac{L_2 H}{L'_2 H'}. \quad (4.2)$$

Si la membrane est libre de se déformer dans les directions de L_2 et de H , les élongations dans ces deux directions doivent être les mêmes [22]. Cette condition minimise l’énergie de déformation. Ceci est une conséquence de la forme incurvée vers le haut de la courbe contraintes-déformations. Nous pouvons donc écrire :

$$\lambda_2 = \lambda_3 = \frac{L'_2}{L_2} = \frac{H'}{H}, \quad (4.3)$$

et donc :

$$\lambda_1 = \lambda_2^{-2}. \quad (4.4)$$

Dans notre cas, nous avons $\lambda_1 = 1.4133$ et $\lambda_2 = 0.8412$, ce qui vérifie bien la relation précédente.

Une autre vérification que nous pouvons faire est que le **module de Young** peut être approximée par la constante C_1 multipliée par 6.

Si nous appliquons un effort total F_1 à l’extrémité droite, tout en attachant l’extrémité gauche, nous pouvons calculer la section de la membrane déformée par :

$$A = \lambda_2 \lambda_3 A_{t_0} = \frac{1}{\lambda_1} A_{t_0}. \quad (4.5)$$

Si les déformations restent petites, nous avons

$$\sigma_{kk} = E \epsilon_{kk} = E(\lambda_{kk} - 1), \quad (4.6)$$

ou encore

$$F_k = EA(\lambda_{kk} - 1). \quad (4.7)$$

Nous pouvons donc calculer le module de Young apparent par :

$$E = \frac{F_1}{A(\lambda_1 - 1)} = \frac{F_1 \lambda_1}{A_{t_0}(\lambda_1 - 1)}. \quad (4.8)$$

Le tableau suivant donne les valeurs de E obtenues pour une membrane de longueur 5 m, de largeur 1 m, d'épaisseur 10 mm et de coefficients C_1 et C_2 égaux à 10^5 Pa et 0 Pa, pour plusieurs intensités de la force F_1 :

F_1 [N]	L [m]	λ_1 [—]	E [MPa]
-100	4.92318	0.98462	0.640195
-10	4.99220	0.99844	0.640355
-1	4.99922	0.999844	0.640269
1	5.00078	1.000156	0.641126
10	5.00785	1.00157	0.637942
100	5.07935	1.01587	0.640120
1000	5.92164	1.18433	0.642511

Nous voyons donc que l'approximation est justifiée mais qu'il serait plus précis d'utiliser $E = 6.4 C_1$. Les forces négatives correspondent à une compression. Si nous n'introduisons aucune modèle de froissement permettant de supprimer les contraintes de compression dans les éléments, et qu'il n'y a aucune perturbation transversale, la membrane entre en compression.

1.2 Membrane en mouvement

Une deuxième observation que nous pouvons faire réside dans la dynamique du problème. Sur la figure 4.4, nous pouvons voir la déformation de la membrane dans les premiers instants de l'application de la force. Les bords de gauche et de droite sont les premiers à subir une déformation importante. Il y a donc un phénomène d'inertie qui donne lieu à des phases intermédiaires intéressantes au cours desquelles différents modes propres de la membrane sont excités. Les oscillations qui en résultent fournissent des déformations particulières, comme celle présentée à la figure 4.5.

Il est difficile de vérifier l'exactitude du mouvement sans disposer de résultats expérimentaux. Toutefois, nous pouvons faire une comparaison intéressante avec un cas théorique des *petites déformations* en gardant en mémoire que ces résultats doivent être nuancés.

Si nous considérons que notre membrane peut vibrer en traction-compression, nous pouvons la comparer à une barre. Prenons par exemple une membrane de longueur 5 m et de section $1 \text{ m} \times 0.01 \text{ m}$ avec des coefficients C_1 et C_2 égaux à 10^5 Pa et 10^4 Pa. Pour cette membrane et pour de petites déformations, nous avons vu que le module

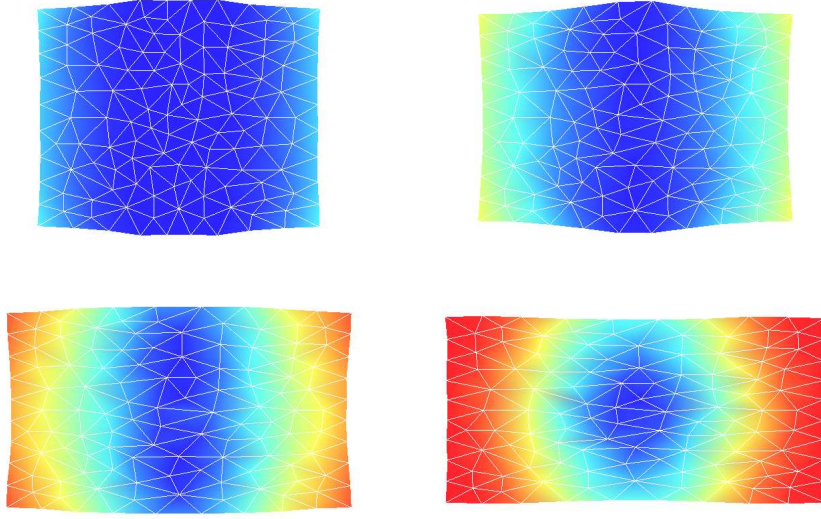


FIG. 4.4 – Quelques-unes des premières images de la déformation du carré.

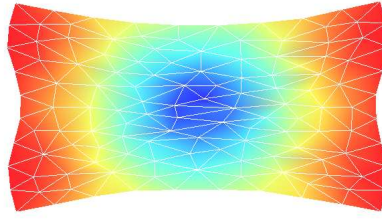


FIG. 4.5 – Exemple de déformation subie avant la stabilisation.

de Young pouvait être approximée à la valeur $6.4 \cdot 10^5 \text{ Pa}$. Les fréquences propres de la barre de même géométrie et de module équivalent sont :

$$f_n = \frac{1}{4} \sqrt{\frac{EA}{\rho AL^2}} (2n - 1) \simeq \frac{1}{4} \sqrt{\frac{6.4C_1A}{\rho AL^2}} (2n - 1). \quad (4.9)$$

La première de ces fréquences vaut 1.265 Hz . Si la membrane est étirée initialement de 1 %, le graphe du déplacement d'un nœud à l'extrémité est le graphe de gauche de la figure 4.6. Nous pouvons observer les oscillations principales ainsi qu'une sinusoïdale parasite provenant d'un mode secondaire. Ce mode provient des contraintes transversales provoquées par l'étirement et il peut être supprimé en donnant une elongation transverse λ_2 calculée via l'équation (4.4), ce qui nous donne le second graphe.

Pour être certain que nos résultats ne seront pas perturbés par un mode impliquant des déplacements transverses, nous fixons tous les nœuds dans cette direction. De

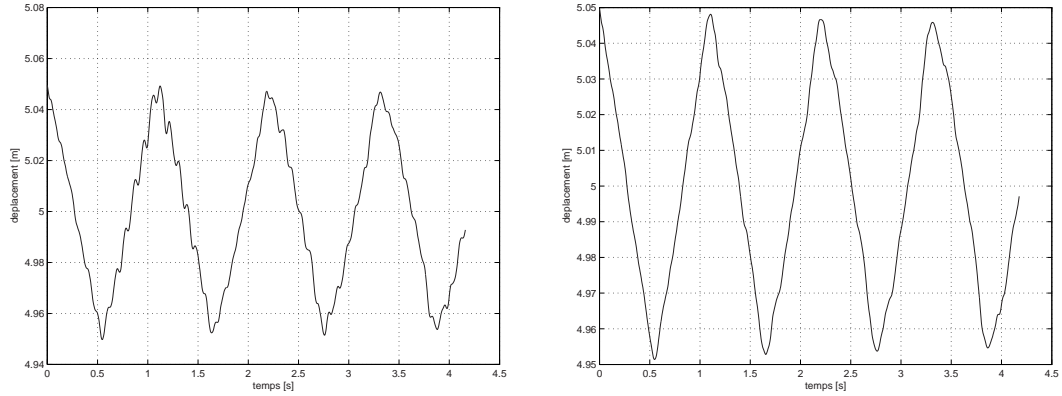


FIG. 4.6 – Evolution du déplacement de l'extrémité libre de la membrane avec étirement simple (à gauche) et étirement accompagné d'un raccourcissement transversal (à droite).

nouveaux tests nous donnent le module de Young apparent dans cette configuration : sous l'effort F_1 , la longueur au repos, l'élongation λ_1 et le module de Young apparent valent :

F_1 [N]	L [m]	λ_1 [–]	E [MPa]
-100	4.942647	0.988529	0.8618
-10	4.994172	0.998834	0.8569
-1	4.999416	0.999883	0.8561
1	5.000583	1.000117	0.8577
10	5.005848	1.001170	0.8557
100	5.059442	1.011888	0.8511

Deux constatations sont à faire. Premièrement, le module est plus élevé en compression qu'en traction. Ensuite, il est plus grand et sa variation est plus importante que pour les tests réalisés précédemment, ce qui se comprend facilement, vu la suppression d'un grand nombre de degrés de liberté et le fait qu'une augmentation de la longueur de l'objet ne puisse plus être compensée que par une diminution de son épaisseur.

Le graphe du déplacement de l'extrémité pour cette nouvelle situation est celui de la figure 4.7, où les déplacements ont été pris suffisamment petits ($\lambda_1 = 1.001$) afin de pouvoir considérer un module de Young constant : $E = 0.8567$ MPa (variation de 0.1 % maximum).

Nous observons tout d'abord que la courbe n'a pas l'allure d'une sinusoïdale, contrairement ce à quoi nous pourrions nous attendre. En observant la déformée, nous pouvons nous rendre compte que les éléments situés à l'extrémité libre accélèrent dans les premières itérations et se déplacent à vitesse constante durant l'oscillation. L'explication est que, en $t = 0$, les nœuds extrêmes sont les seuls à ne subir de forces internes que dans un seul sens, puisque les autres se trouvent entre des éléments étirés et qu'ils subissent donc des forces internes de directions opposées. Lorsque les premiers

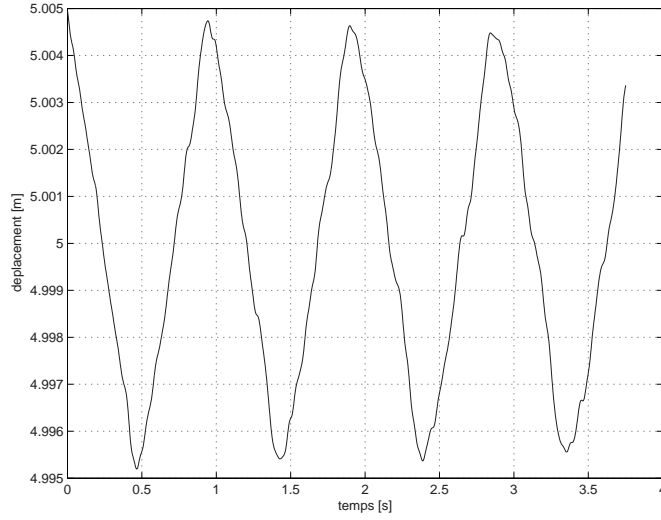


FIG. 4.7 – Evolution du déplacement de l’extrémité libre de la membrane en fixant les nœuds dans le sens transverse.

nœuds sont en mouvement, la deuxième rangée de nœuds subit des forces internes non-symétriques qui vont les mettre en mouvement et ainsi de suite. Une onde va donc se déplacer le long de la membrane, rebondir à l’autre extrémité, et revenir à son point de départ ; le tout à la fréquence propre de la membrane.

Si la membrane réagissait de la même manière qu’une barre soumise à de petites déformations, nous devrions obtenir une fréquence de (équation (4.9)) :

$$f_1 = \frac{1}{4} \sqrt{\frac{0,8567 \cdot 10^6}{1000 \cdot 5^2}} = 1,46 \text{ Hz}. \quad (4.10)$$

La fréquence que nous obtenons est moins élevée. Nous pouvons voir sur le graphe de la figure 4.7 que la période est d’environ 0.9 secondes, ce qui correspond à une fréquence de 1.11 Hz. La différence est de 30 % environ. Nous voyons donc que les deux éléments comparés, la barre et la membrane, ont des fréquences dont l’ordre de grandeur est fort proche. Des tests menés avec d’autres valeurs de L , l , H , C_1 , C_2 et λ_1 indiquent toujours qu’un rapport proche de 30 % sépare les deux modèles.

La différence obtenue peut être la conséquence de plusieurs facteurs. Tout d’abord, la section de la membrane est variable, son épaisseur change durant les oscillations et notre modèle impose que le matériau soit incompressible. De plus, le modèle de Mooney-Rivlin est un modèle adapté aux grandes déformations et sa précision n’est pas prouvée pour des dilatations très petites. Enfin, nous avons vu que le mouvement des nœuds n’était pas régulier le long de la membrane. Les phénomènes d’inertie décrits plus haut influencent sans nul doute la forme du graphe des déformations et la fréquence calculée.

Sans prouver que les mouvements obtenus modélisent exactement la réalité, les tests précédents et les constatations faites indiquent tout de même que le programme

fournit une bonne approximation du comportement de la membrane dans le temps. Des analyses statiques de problèmes en grandes déformations permettraient de compléter ces tests et d'aller plus loin dans la vérification du programme. Dans les sections qui suivent, nous vérifions le calcul des déformations statiques d'un cylindre et d'une sphère en grandes déformations.

2 Cylindre infini

Le cas du cylindre infini soumis à une pression intérieure est intéressant car il est possible de comparer facilement les résultats numériques et analytiques. En effet, un seul paramètre géométrique, le rayon, permet de décrire la déformation. De plus, il est aisé de trouver une solution analytique au problème statique.

Maillage utilisé

Pour représenter un cylindre infini, il suffit de représenter une fine tranche du cylindre en lui appliquant les conditions aux frontières adéquates. La figure 4.8 représente le maillage d'une tranche. La hauteur représentée est celle d'un élément car il serait inutile de calculer simultanément plusieurs étages.

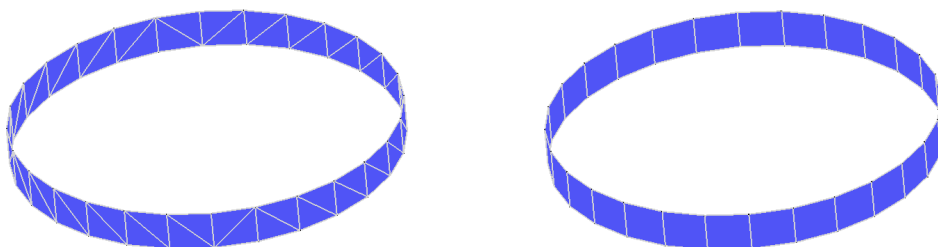


FIG. 4.8 – Maillages du cylindre, par triangles et par quadrangles.

Le long d'un cylindre infini, la déformation axiale doit être la même sur chaque tranche (pas de site favorisé) et le cylindre ne peut s'allonger ni se rétrécir. Chaque tranche présente donc une hauteur fixe. Les conditions aux frontières utilisées ici sont donc un blocage de tous les nœuds dans le sens de l'axe du cylindre.

Vérification de la relation (p, R)

La relation théorique entre pression et rayon, en statique, s'exprime [6] par :

$$p = \frac{2C_1 H}{R_0} (1 + \alpha) \left(1 - \frac{1}{\lambda^4}\right), \quad (4.11)$$

où C_1 et C_2 sont les constantes du modèle de Mooney-Rivlin, H est l'épaisseur initiale de la membrane, R_0 est le rayon initial, α est le rapport $\frac{C_2}{C_1}$ et λ est le rapport $\frac{R}{R_0}$.

Notons que p peut, d'une manière plus générale, désigner une différence de pression entre l'intérieur et l'extérieur du cylindre.

La relation analytique est représentée par la courbe bleue sur la figure 4.9. Nous

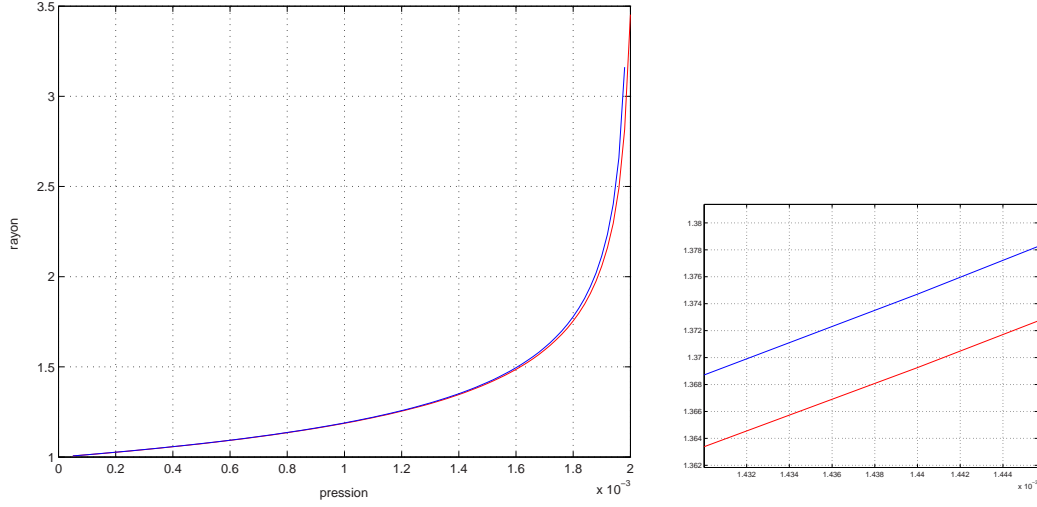


FIG. 4.9 – Comparaison des solutions numérique et analytique pour l'élongation du cylindre infini (exprimée en mètres) en fonction de la pression (en pascals).

remarquons que la courbe présente deux zones : une zone quasi-linéaire pour des pressions faibles et une zone non-linéaire tendant vers une asymptote. Cette non-linéarité est caractéristique de matériaux tels que le caoutchouc et certains polymères. Nous pouvons l'observer très simplement en gonflant un ballon de baudruche à la bouche : avant qu'il ait atteint un certain volume, il faut souffler de plus en plus *fort* pour le gonfler. Une fois la zone linéaire franchie, une faible augmentation de pression conduit à une augmentation de volume importante, et la difficulté devient alors de fournir un *volume* de gaz suffisant au ballon pour achever de le remplir.

Au-delà d'une certaine pression, le cylindre gonfle indéfiniment. Avec les paramètres choisis pour tester le cylindre infini, à savoir $H = 1 \text{ mm}$, $C_1 = 1 \text{ Pa}$, $R_0 = 1 \text{ m}$ et $\alpha = 0$, la pression critique est de 0.002 kg/m^2 .

La non-linéarité théorique est exprimée dans notre modèle par la loi de comportement choisie (équation 1.81) :

$$W = C_1(I_1 - 3) + C_2(I_2 - 3), \quad (4.12)$$

puisque les invariants I_1 et I_2 sont des fonctions non-linéaires des élongations principales (équations 1.74 et 1.75) :

$$I_1 = \lambda_1^2 + \lambda_2^2 + \lambda_3^2, \quad (4.13)$$

$$I_2 = \lambda_1^2 \lambda_2^2 + \lambda_2^2 \lambda_3^2 + \lambda_1^2 \lambda_3^2. \quad (4.14)$$

Pour fournir une courbe comparable à la courbe théorique, nous avons écrit un programme exécutant notre code pour une série de pressions, en utilisant le maillage triangulaire de 56 éléments représenté à la figure 4.8. Les résultats sont représentés sur la figure 4.9 par la courbe rouge.

Nous observons que les écarts sont très faibles dans la zone linéaire (moins de 1 %), et qu'ils deviennent sensiblement plus importants pour des pressions plus élevées (1 à 3 % et jusque 13 % dans la zone quasi-verticale du graphe). Des tests réalisés avec des maillages plus fins montrent que le raffinement du maillage réduit ces écarts.

Les résultats obtenus avec le maillage de quadrangles de la figure 4.9 sont identiques à ceux du maillage de triangles. La concordance des résultats est semblable à celle obtenue avec des mailles triangulaires.

3 Membrane précontrainte

Comme expliqué dans la section I.2.3.6, la combinaison de conditions aux frontières et de déformations initiales permet d'envisager une notion utile dans l'étude de cas pratiques de membranes : la précontrainte. Si nous imposons une déformation initiale à une membrane et que nous bloquons ses extrémités, une traction subsistera. La figure 4.10 montre un cylindre attaché à ses extrémités, et étiré de 40 %.

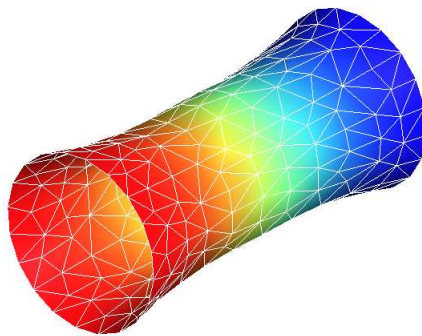


FIG. 4.10 – Cylindre précontraint en configuration déformée.

Le cas du cylindre attaché par ses extrémités nous a permis de visualiser la déformation d'un objet 3D de manière assez simple. Il nous a également permis de tester l'imposition de conditions aux limites et l'intervention d'un amortissement.

Quelques images du cylindre en configuration initiale et déformée sont présentées à la figure 4.11. La déformation est due à une pression interne appliquée subitement au temps t_0 et maintenue constante.

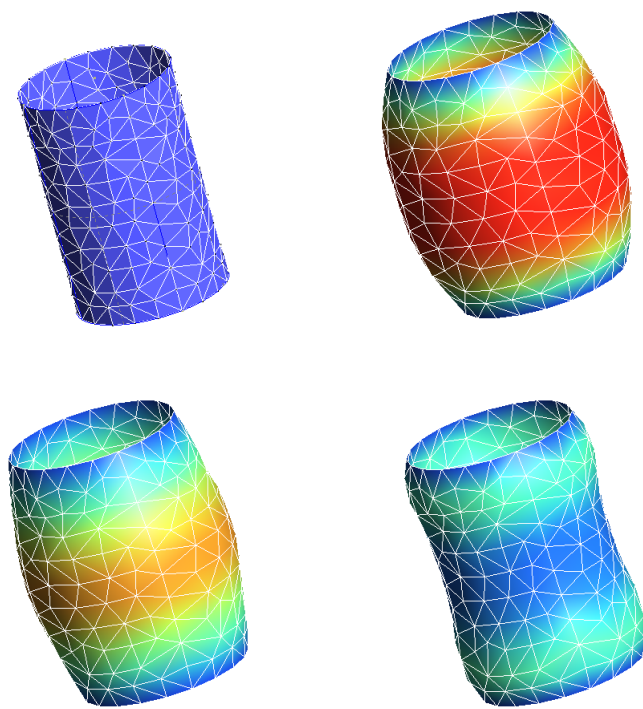


FIG. 4.11 – Déformations successives du cylindre attaché par ses extrémités.

4 Sphère

Vérification de la relation (\mathbf{p}, \mathbf{R})

La vérification du programme au travers de l'exemple de la sphère est plus compliquée que pour le cylindre infini car les sources littéraires sont plus rares et surtout, elles ne sont pas toutes en accord. Il est donc très important de porter un regard critique sur nos sources.

Nous présentons ici les principales formulations rencontrées dans la littérature :

– Formulation 1 [23] :

$$P = 2n_{seg}KT \left(\frac{H_0}{R_0} \right) \left(\left(\frac{R_0}{R} \right)^1 - \left(\frac{R_0}{R} \right)^7 \right), \quad (4.15)$$

où $n_{seg}KT = C1$.

– Formulation 2 [24] :

$$P = \frac{2C1H_0}{R_0} \left(\frac{R_0}{R} - \left(\frac{R_0}{R} \right)^7 \right) \left(1 - \frac{C1}{C2} \left(\frac{R}{R_0} \right)^2 \right). \quad (4.16)$$

– Formulation 3 [25] [26] :

$$P = \frac{4C1H_0}{R_0} \left(\frac{R_0}{R} - \left(\frac{R_0}{R} \right)^7 \right) \left(1 + \frac{C2}{C1} \left(\frac{R}{R_0} \right)^2 \right). \quad (4.17)$$

Ces formulations ne sont pas complètement équivalentes. Il apparaît dans l'ensemble que la pression de la sphère dépend de $\left(\frac{R_0}{R} - \left(\frac{R_0}{R} \right)^7 \right)$. Nous avons pu remarquer lors de l'essai de notre programme, que celui-ci suivait bien ce comportement (figure 4.12).

Signalons que, contrairement au cas du cylindre infini, la validation se fait au travers de la pression en fonction du rayon, ce qui signifie que nous faisons varier le rayon et que nous trouvons la pression correspondante. Cela permet de trouver des positions stables au-delà de la déformation critique que nous avons observée avec le cylindre infini, mais en appliquant une pression qui décroît avec le rayon.

Nous voyons que, malgré les différences de pressions limites, les rayons critiques restent les mêmes et les allures se superposent à une constante près, qui varie selon les sources.

Remarquons que dans ce cas, les valeurs de la pression critique varient très peu avec le raffinement du maillage (0.2 % si nous divisons la taille de la maille par deux), et qu'il est possible d'obtenir une très bonne approximation de la courbe avec un maillage relativement grossier.

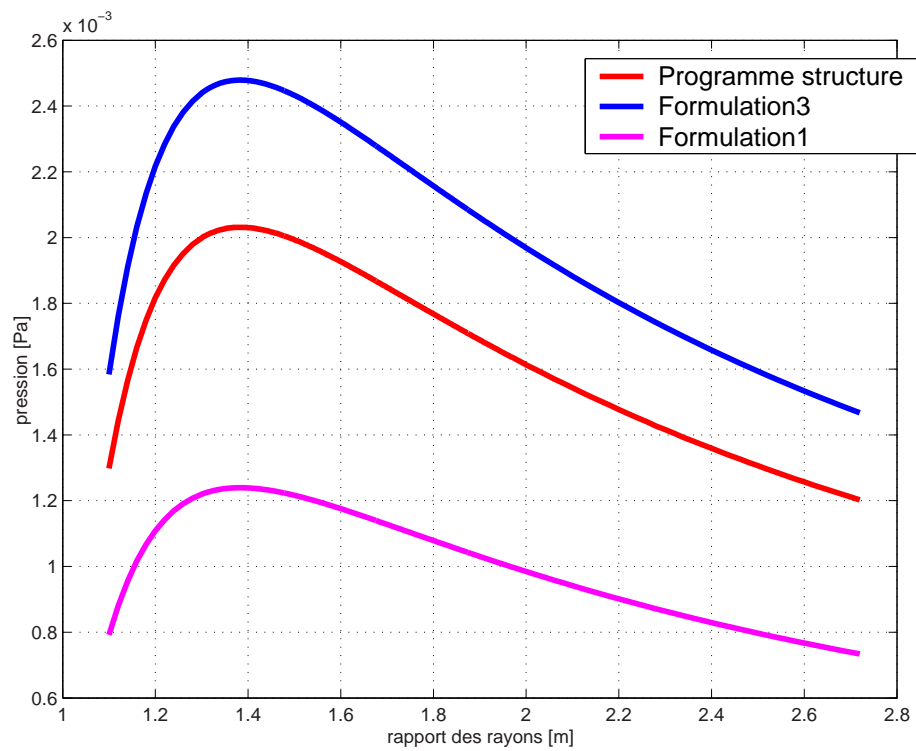


FIG. 4.12 – Comparaison du comportement de la sphère entre certaines théories trouvées dans la littérature et les résultats de notre code de calcul.

5 Déformations extrêmes

Dans des situations où les forces sont telles que la membrane se retourne entièrement, qu'elle se froisse ou qu'elle subit de très grandes déformations, le code se montre assez robuste. Sur la figure 4.13, nous voyons une membrane dont certaines parties sont incurvées vers l'intérieur du volume. Sous une certaine pression, ces parties se renversent brutalement sans déstabiliser le programme.

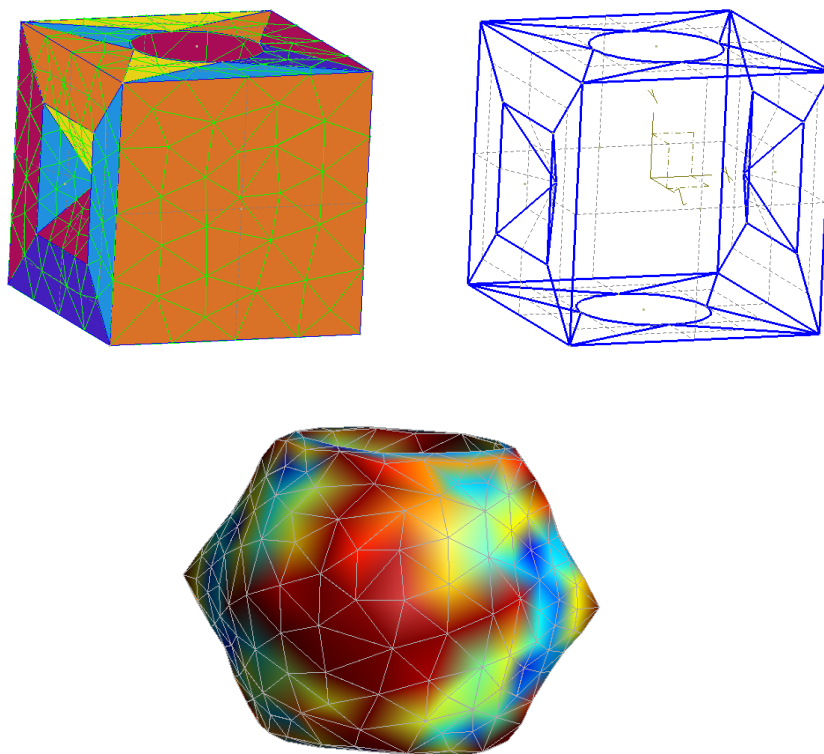


FIG. 4.13 – Membrane incurvée et sa déformée sous une pression élevée.

La figure 4.14 montre les déformées successives d'une membrane cubique dont deux faces sont laissées ouvertes. La membrane est attachée autour de l'une des deux faces ouvertes et une pression intérieure relativement élevée lui est appliquée brutalement. Nous pouvons voir la membrane s'élargir considérablement, puis se retourner et enfin revenir vers sa position de départ du fait de ses propriétés hyperélastiques. Nous remarquons que la membrane se froisse entre deux éléments du maillage. Cela est dû au fait que nous n'avons pas introduit de modèle de froissement dans les équations même. L'absence de froissement à l'intérieur d'un élément ne provoque qu'une faible erreur pour autant que le maillage soit suffisamment raffiné.

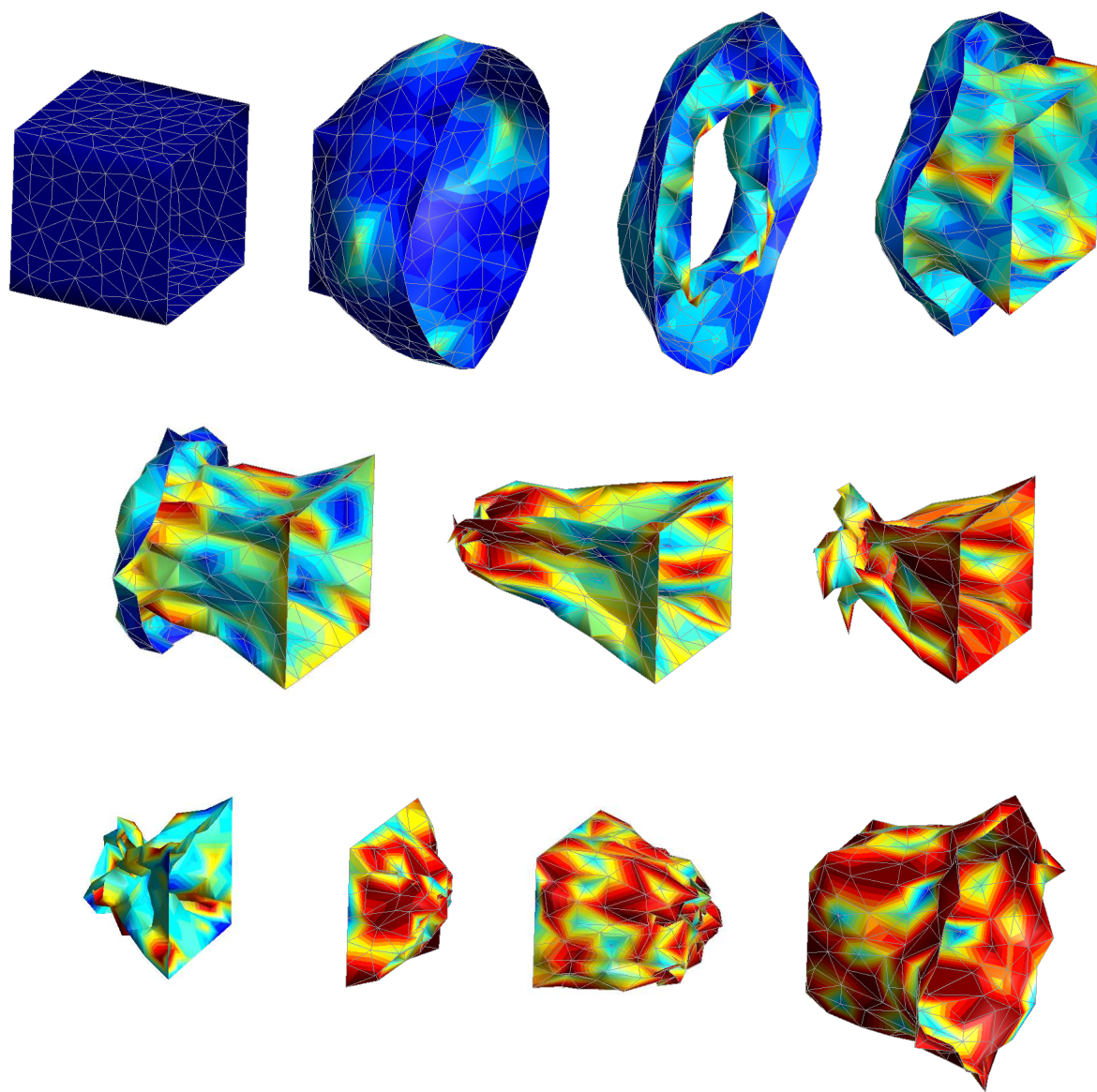


FIG. 4.14 – Evolution de la déformée d'un cube soumis à une pression interne élevée.

6 Applications diverses

Feuille attachée par un côté

La figure 4.15 montre quelques positions occupées par une feuille de caoutchouc sur laquelle une pression constante est appliquée, et attachée par un de ses cotés seulement. Les valeurs prises pour C_1 et C_2 sont 0.13 MPa et 0.095 MPa .

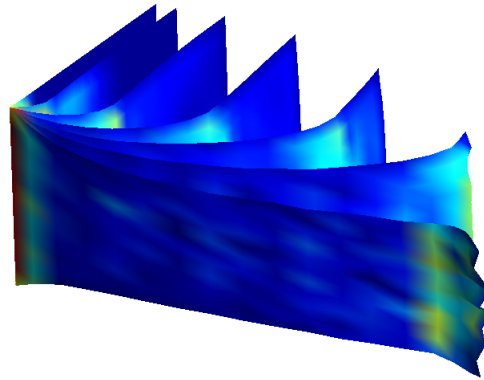


FIG. 4.15 – Superposition de vues, à des temps différents, d’une feuille attachée par un côté et soumise à une pression constante sur l’une de ses faces.

Nous observons que l’augmentation de vitesse due à la pression provoque également une augmentation de la force centrifuge et donc une déformation de la feuille. Comme vu plus haut, l’allongement de la membrane dans une direction va de pair avec un raccourcissement dans l’autre direction.

Membranes gonflables

Un type de problème que le programme peut modéliser sans faire intervenir de couplage avec un programme fluide est le gonflement d’une structure à l’intérieur de laquelle on maintient une pression constante, à condition que les effets de la vitesse et de la viscosité du fluide soient négligeables (ballons, procédés de mise en forme thermoplastique par soufflage, etc). Le cylindre infini en est un exemple. Les figures 4.16, 4.17 et 4.18 montrent quelques exemples de gonflements de membranes que nous avons pu obtenir.

Ces quelques exemples terminent la partie de ce rapport relative au programme de calcul de membranes.

Nous avons réalisé un programme permettant de modéliser les déformations et les

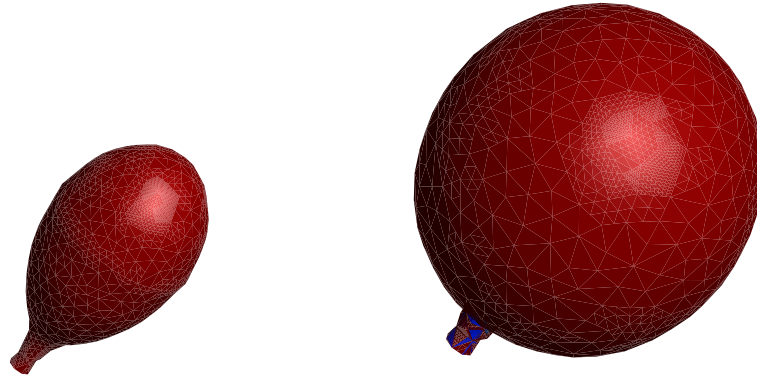


FIG. 4.16 – Un même ballon à plat et gonflé ($C_1 = 0.1 \text{ MPa}$, $C_2 = 0.01 \text{ MPa}$).

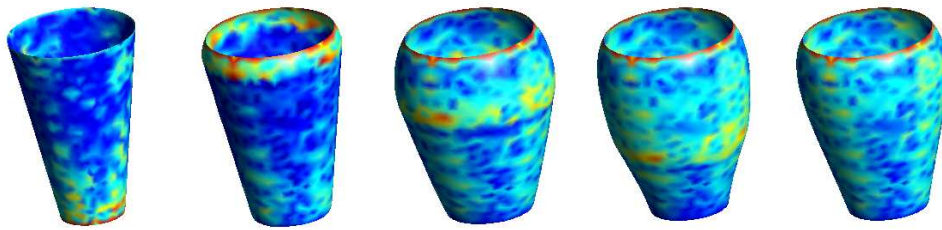


FIG. 4.17 – Plusieurs phases du gonflement d'un cône ($C_1 = 1.0 \text{ MPa}$, $C_2 = 0.05 \text{ MPa}$).

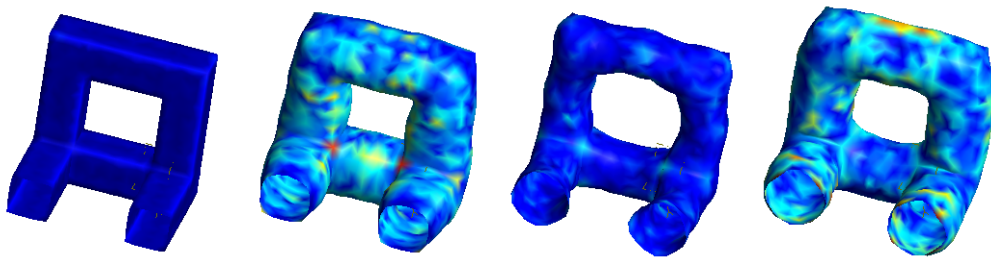


FIG. 4.18 – Gonflement et oscillation d'une structure quelconque ($C_1 = 0.5 \text{ MPa}$, $C_2 = 0 \text{ MPa}$).

déplacements d'une membrane quelconque au cours du temps. Si les caractéristiques de la membrane sont connues et qu'une pression constante lui est appliquée, nous pouvons la discrétiser et étudier son comportement en observant les déplacements, vitesses et

accélérations de ses noeuds. Nous pouvons également visionner sa déformation via *Gmsh*.

Dans la suite de ce rapport, nous expliquons les travaux que nous avons menés afin de réaliser un couplage du programme avec un programme de calcul de fluides, ainsi que les évolutions que nous avons apportées à notre code pour atteindre notre objectif.

Deuxième partie

Couplage avec un programme de
calcul de fluides

Chapitre 5

Description du programme fluide

Pour modéliser le comportement du fluide au contact de la membrane, nous avons recours à un programme nommé *Aero-F* [27], développé et validé par Philippe Geuzaine¹. Le développement d'un tel programme est en effet une tâche ardue pouvant à elle seule demander plusieurs années de travail. Par ailleurs, ce programme comporte déjà un module destiné à faciliter le couplage fluide-structure.

Dans le contexte présent, l'utilisation d'un programme existant ne peut se faire sans avoir une certaine compréhension des concepts fondamentaux liés à son implémentation. C'est pourquoi nous avons cherché à savoir quel était le modèle mathématique et quelles étaient les méthodes numériques employées par son concepteur.

1 Modèle mathématique

1.1 Equations fondamentales

Avant² de pouvoir comprendre une solution numérique, il est nécessaire de connaître le modèle mathématique, c'est-à-dire les équations fondamentales qui gouvernent le type d'écoulement que nous modélisons.

Les équations fondamentales de la mécanique des fluides sont la traduction mathématique de trois principes physiques fondamentaux :

- la conservation de la masse : l'équation traduisant ce principe est appelée *équation de continuité*,
- la seconde loi de Newton $F = ma$: appliquée à un fluide, elle fournit une équation vectorielle connue sous le nom d'*équation du moment*,
- la conservation de l'énergie : ce principe est le premier principe de la thermo-

¹Docteur en sciences appliquées, chercheur au CENAERO à Gosselies.

²Cette section est en partie inspirée du rapport de mémoire d'Emilie Marchandise et de Marie Goffinet [28].

dynamique et l'équation mathématique en découlant est appelée *équation de l'énergie*.

Pour un écoulement *tridimensionnel transitoire* et pour un fluide *visqueux* de viscosité ν , *compressible*, les équations fondamentales peuvent s'écrire en coordonnées cartésiennes [29] :

Equation de continuité :

$$\frac{\partial \rho}{\partial t} + \frac{\partial u_i}{\partial x_i} = 0 \quad (5.1)$$

Equations de conservation de la quantité de mouvement :

$$\frac{\partial}{\partial t}(\rho u_j) + \underbrace{\frac{\partial}{\partial x_i}(\rho u_i u_j)}_{\text{convection}} = - \underbrace{\frac{\partial p}{\partial x_i}}_{\text{gradient de pression}} + \underbrace{\frac{\partial^2}{\partial x_i^2}(\rho \nu u_j)}_{\text{diffusion}} + \underbrace{g_j}_{\text{forces volumiques}} \quad (5.2)$$

Equation de conservation de l'énergie :

$$\frac{\partial}{\partial t}(\rho c T) + \nabla \cdot (\rho c T \vec{V}) = -\nabla \cdot \vec{q} + \vec{\tau}_{ij} : \nabla V \quad (5.3)$$

Précisons les termes intervenant dans l'équation de l'énergie :

- c est la chaleur spécifique,
- τ le tenseur des contraintes visqueuses,
- q le flux de chaleur,
- T la température.

Habituellement, le système composé de l'équation vectorielle de moment et de l'équation de continuité est appelé *équations de Navier-Stokes*, même si historiquement, les équations de Navier-Stokes ne désignaient que les équations du moment.

Les équations d'Euler sont semblables à des équations de Navier-Stokes dans lesquelles nous aurions imposé une viscosité nulle.

Aero-F utilise les deux modèles, Navier-Stokes et Euler, sous leur forme adimensionnalisée, pour des fluides compressibles, selon que nous voulions modéliser un écoulement visqueux ou non-visqueux. Le fait que ces équations soient adimensionnalisées signifie que leurs paramètres seront des nombres adimensionnels tels que le nombre de Reynolds et le nombre de Mach.

Note sur les fluides compressibles

Un³ fluide est dit compressible s'il a la possibilité de changer de volume sous l'effet de la pression. C'est en particulier le cas des gaz. Cependant, lors de l'étude de l'écoulement autour d'une voiture par exemple, les effets de compressibilité de l'air ne sont pas pris en compte et l'étude est faite dans le cadre d'un modèle incompressible.

³Note rédigée par Jean-François Remacle.

Ainsi, Les effets de compressibilité ne sont pas uniquement liés à la nature intrinsèque du fluide mais également à l'écoulement. Dans le cas d'un écoulement permanent, le long d'une ligne de courant (i.e. 1D), l'équation d'Euler (quantité de mouvement) nous donne :

$$u du + \frac{dp}{\rho} = 0, \quad (5.4)$$

u est la vitesse, p la pression et ρ la densité.

La vitesse du son est définie comme

$$c^2 = \frac{dp}{d\rho}. \quad (5.5)$$

Nous pouvons donc écrire

$$u du + \frac{dp}{d\rho} \frac{d\rho}{\rho} = u du + c^2 \frac{d\rho}{\rho} = 0. \quad (5.6)$$

Nous avons donc

$$\frac{d\rho}{\rho} = -Ma^2 \frac{du}{u}. \quad (5.7)$$

L'accélération (décélération) du fluide va donc créer une diminution (augmentation) de la densité du fluide si le nombre de Mach $Ma=u/c$ est significatif. Typiquement, dans des écoulements environnementaux, Ma est de l'ordre de $1/30 - 1/100$ et les effets de compressibilité sont négligeables.

1.2 Formulation ALE

Les équations précédentes sont écrites en formulation eulérienne. Sous cette forme, elles sont adaptées à l'étude d'un écoulement se produisant entre des frontières fixes mais ne permettent pas de s'attaquer à un problème de couplage fluide - parois mobiles.

Pour ce genre de problème, il est préférable de se tourner vers une formulation lagrangienne du mouvement. Contrairement au cas précédent, il est nécessaire que notre volume de contrôle évolue avec le fluide. Les particules y sont renseignées par des coordonnées matérielles. Cette description pose donc problème lorsqu'il s'agit de suivre une particule sur une période relativement longue.

Une formulation hybride, appelée formulation *ALE* pour « *Arbitrary Lagrangian Eulerian* », permet de surpasser les problèmes liés à chacune des formulations. Elle permet de passer d'une manière continue des coordonnées spatiales aux coordonnées matérielles et inversement [30].

Dans une formulation ALE, une coordonnée hybride est créée. Cette coordonnée imaginaire est supposée constante lorsqu'elle suit un point du maillage. Ces coordonnées équivalent aux coordonnées lagrangiennes si les points du maillage suivent les mouvements des particules. Si le maillage est fixe, ces coordonnées sont équivalentes

aux coordonnées spatiales. On suppose qu'il existe deux transformations permettant de passer du système de coordonnées hybrides aux coordonnées spatiales et inversement. Les équations fondamentales exprimées plus haut peuvent alors être réécrites en fonction de ces coordonnées.

C'est cette dernière formulation des équations qui a été implémentée pour *Aero-F*.

1.3 Turbulence

Observons⁴ la fumée d'une cigarette montant dans l'air calme (figure 5.1) : que voit-on ? Sur quelques dizaines de centimètres, la fumée s'élève tranquillement et les



FIG. 5.1 – Fumée dégagée dans un milieu calme.

particules de fumée nous paraissent suivre des trajectoires régulières et uniformes ; puis sur une dizaine de centimètres environ, cet ordonnancement se disloque au profit de quelques tourbillons d'une taille comparable à la largeur de la colonne de fumée ; ils évoluent et se déforment en s'élevant pour brusquement laisser place à un mouvement tellement désordonné que nous ne pouvons plus suivre à l'œil nu la trajectoire d'une particule.

Dans la zone basse, l'écoulement ordonné est qualifié de *laminaire* ; dans la plus haute, il est dit *turbulent*, tandis que la zone intermédiaire est appelée *zone de transition*. Dans ces trois zones, les conditions dans lesquelles s'effectue l'écoulement sont voisines, et pourtant les écoulements sont très différents.

La variable représentant le mieux la possibilité d'apparition d'un écoulement turbulent est le nombre de *Reynolds*, nombre adimensionnel noté Re . Il exprime le rapport entre les forces d'inertie des particules et les forces visqueuses. Si nous reprenons les équations de Navier-Stokes (5.2), nous pouvons constater qu'un terme représente la diffusion et un autre la convection. Le nombre de Reynolds s'obtient en écrivant le

⁴Cette section est en partie inspirée du rapport de mémoire d'Emilie Marchandise et de Marie Goffinet [28].

rapport entre ces deux termes :

$$\frac{\text{convection}}{\text{diffusion}} = \frac{\nabla_i(\rho u_i u_j)}{\rho \nu \nabla_i^2 u_j} \propto \frac{\rho \frac{U^2}{L}}{\rho \nu \frac{U}{L^2}} = \frac{UL}{\nu} = Re. \quad (5.8)$$

La turbulence apparaît dans les écoulements dont le nombre de Reynolds est élevé. Elle a souvent comme origine le développement d'une instabilité d'un écoulement laminaire. Lorsque le nombre de Reynolds dépasse une valeur critique, les petites perturbations deviennent instables et leur croissance rapide produit la transition vers la turbulence.

Pour modéliser la turbulence, plusieurs modèles existent. Nous n'entrerons pas dans les détails mais nous présenterons sommairement les plus répandus :

- On effectue depuis quelques années des **simulations directes** ou **DNS** (« *Direct Numerical Simulation* ») de la turbulence, mais uniquement dans des configurations géométriques simples et pour des nombres de Reynolds peu élevés. Comme son nom l'indique, une simulation directe résout « directement » les équations de Navier-Stokes sans les approximer. Cette approche semble la plus évidente du point de vue conceptuel. Cependant, les temps de calcul requis sont considérables et le nombre de mailles nécessaires gigantesque.
- Dans une autre catégorie, les méthodes les plus courantes sont basées sur une approche **statistique**. On décompose alors toutes les variables qui décrivent l'écoulement en une valeur moyenne et une fluctuation ($\phi = \bar{\phi} + \phi'$). Ceci s'appelle la décomposition de Reynolds. Cette technique permet d'écrire un système d'équations qui décrit le champ d'écoulement moyen.
- Une méthode de modélisation de la turbulence qui semble être un bon compromis entre le DNS et l'approche statistique est la **simulation des grandes échelles** ou **LES** (« *large eddy simulation* »), ne modélisant que les turbulences de grande échelle (souvent les plus intéressantes) et utilisant un modèle de fermeture pour les plus petites.

Aero-F met à notre disposition deux modèles statistiques de turbulence, que nous nous contenterons de citer :

- modèle Spalart-Allmaras,
- modèle $k - \epsilon$.

Il est également possible de n'utiliser aucun modèle. Ajoutons que le programme peut modéliser des écoulements laminares également.

2 Méthodes de discrétisation

Cette section et la suivante sont destinées à donner une explication sommaire à propos des méthodes utilisées pour discrétiser le domaine et les équations et procéder

à leur résolution. Une recherche des différentes méthodes à disposition pour ce genre de problème et une investigation plus poussée des techniques utilisées dans *Aero-F* ne manquerait pas d'intérêt. Néanmoins, notre temps étant limité, nous avons préféré laisser ces recherches de côté.

2.1 Technique de discrétisation spatiale

La technique de discrétisation utilisée est une méthode mixte éléments finis et volumes finis. La technique des volumes finis est une méthode de résidus pondérés dans laquelle les fonctions de pondération sont égales à l'unité dans des volumes finis donnés et nulles partout ailleurs. Elle est à l'origine de la majorité des codes de calculs en géométrie cartésienne. Les volumes sont centrés aux nœuds (figure 5.2).

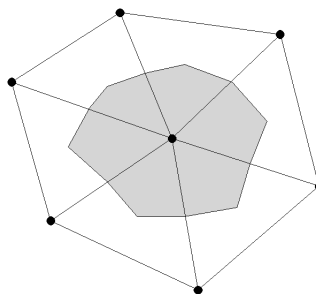


FIG. 5.2 – Surface centrée sur un nœud en 2D.

Le maillage du volume est un maillage non-structuré de tétraèdres. Celui-ci doit pouvoir bouger et/ou se déformer lorsqu'une frontière se déplace ou se déforme, comme dans le cas d'une interaction fluide-structure. La génération d'un nouveau maillage à chaque pas de temps est peu économique et prend beaucoup de temps.

Une approche plus économique consiste à faire évoluer le maillage d'un pas de temps à l'autre. C'est ce que fait *Aero-F*. Le programme sépare les composantes rigides et de déformation du mouvement des frontières. Il calcule ensuite les nouvelles positions des sommets des tétraèdres en associant une rigidité à chaque arête et en minimisant l'énergie potentielle de ce système.

2.2 Techniques de discrétisation des équations

L'une des difficultés de la méthode des volumes finis réside dans l'estimation des flux aux frontières des éléments. L'intégration des équations d'Euler et de Navier-Stokes nécessite une méthode d'interpolation spécifique pour les calculer.

Pour intégrer le terme de convection dans les équations de moment, un schéma de type Roe du second ordre est appliqué. La méthode de Roe définit une linéarisation locale de l'équation d'Euler sur les discontinuités, situées aux interfaces des mailles [30].

Une approximation éléments finis P1 de Galerkin est utilisée pour les flux visqueux, c'est-à-dire la partie diffusive des équations. La méthode de Galerkin consiste à utiliser les fonctions de forme comme poids dans une méthode de résidus pondérés.

Le programme permet de réaliser l'intégration temporelle avec des algorithmes implicites du premier ou du second ordre.

Chapitre 6

Description du couplage

1 Fonctionnement général

Pour calculer une interaction fluide-membrane, une architecture possible serait la résolution des deux systèmes, fluide et structure, dans un même code. Cela nécessiterait l'écriture d'un programme très lourd (dépassant le cadre d'un mémoire) ou une modification profonde de l'un des codes. Par ailleurs, nous perdrons l'avantage de la modularité d'un système de codes indépendants.

L'option que nous avons prise dans ce mémoire est plutôt de réaliser un couplage fluide-structure en associant notre code au programme de calcul des fluides et de les faire travailler en étroite communication autour d'un même problème. Désormais nous appellerons ces programmes *programme fluide* et *programme membrane*.

Concrètement, cela implique au moins les deux éléments suivants :

- un directeur, qui répartit les tâches et gère la communication entre ses deux spécialistes. Le rôle peut être joué par l'un d'entre eux,
- un moyen de communication (passage de paramètres, fichiers, ...).

Dans notre cas, c'est le programme fluide qui exécute le programme structure. Un module permettant d'exécuter du code Matlab lors de chaque itération a été implémenté par Philippe Geuzaine. Il y aura donc un grand nombre d'exécutions du code structure à l'intérieur de l'exécution du programme fluide.

Au niveau du calcul forces-déplacements, deux possibilités de fonctionnement sont envisageables.

- Le programme fluide calcule les forces qui s'appliquent à la membrane et les communique au programme structure, qui calcule ensuite les déplacements. Ceux-ci sont imposés au programme fluide lors de l'itération suivante.
- Le programme fluide calcule les déplacements et la membrane calcule les pressions correspondantes.

Physiquement, on pourrait penser que la deuxième solution est la meilleure, car la masse du fluide est souvent la plus grande et son mouvement aura une inertie importante. Néanmoins, les résultats que nous devons obtenir sont les mêmes et ce choix n'est pas trop important. Afin de ne pas avoir à modifier le programme fluide, conçu pour recevoir des déplacements et renvoyer des forces, nous avons opté pour la première solution. La figure 6.1 montre le schéma de fonctionnement général du couplage.

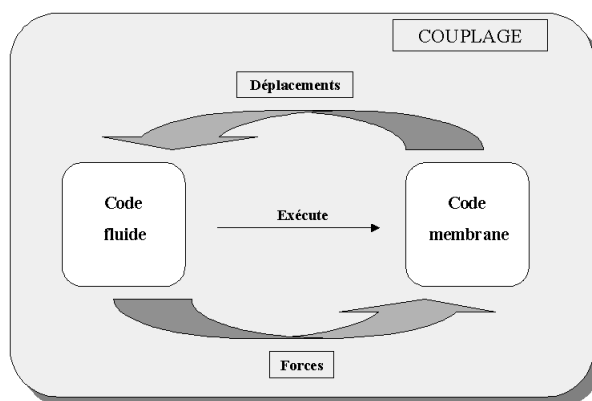


FIG. 6.1 – Schéma général du couplage.

Le moyen de communication entre les codes est l'écriture/lecture de fichiers. Ainsi, le programme fluide va écrire un fichier contenant les forces dans les trois directions de l'espace en chaque nœud de la frontière fluide, et le programme structure va écrire un fichier contenant tous les déplacements des nœuds de la membrane.

Choix des pas de temps

Le programme fluide utilise un algorithme implicite qui, en théorie, assure sa stabilité quel que soit le pas de temps. Le choix de celui-ci n'est donc pas dicté par des impératifs de stabilité, mais plutôt par les déformations éventuelles de la membrane entre deux pas de temps. En effet, si les déformations sont grandes entre les temps t et $t + 1$, la solution fluide sera peu précise puisque calculée dans un premier temps sur la configuration t uniquement. Le programme diminue néanmoins l'erreur en recommençant le calcul avec la nouvelle situation dès le début de l'itération $t + 1$.

Le pas de temps du programme membrane Δt_{mbr} se déduit de la condition de stabilité de l'algorithme explicite utilisé. Si le pas de temps limite est inférieur au pas de temps Δt_{fl} du fluide, plusieurs itérations sur la membrane seront nécessaires à l'intérieur d'une seule itération fluide. Un ratio $\Delta t_{mbr}/\Delta t_{fl}$ que nous avons souvent utilisé est 1/5.

2 Maillages 3D

Un des aspects sur lequel nous avons dû nous pencher pour réaliser le couplage est la création des maillages et leur interprétation par les deux programmes.

2.1 Concordeance 2D et 3D

Pour ne pas compliquer le couplage, il nous a paru préférable d'établir une concordeance parfaite entre le maillage de tétraèdres utilisé par le programme fluide et le maillage de la membrane. Tout nœud de la membrane doit être un nœud du maillage de tétraèdres (figure 6.2). Cela implique de réaliser les maillages en même temps, de sorte que la génération du maillage 3D s'appuie sur le maillage 2D (quand nous écrivons 2D, nous voulons parler de la membrane avec son épaisseur très faible, même si celle-ci peut prendre n'importe quelle géométrie dans les trois dimensions de l'espace).

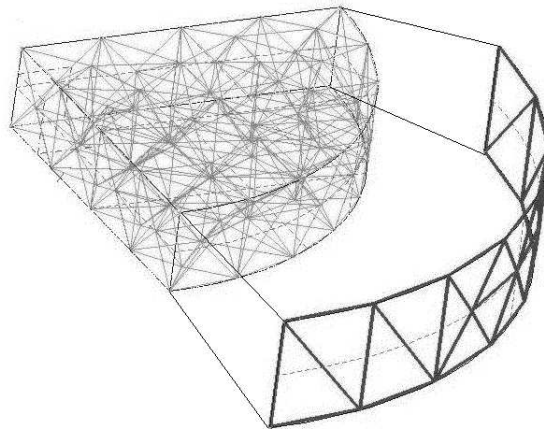


FIG. 6.2 – Maillages de la membrane et du volume de fluide correspondant.

Nous réalisons ces maillages avec le même logiciel que pour les membranes seules, *Gmsh*.

Le maillage *global* comprendra donc des nœuds, des arêtes, des triangles et des tétraèdres. Parmi les nœuds, nous pouvons déterminer trois catégories (figure 6.3) :

1. les nœuds situés sur la membrane,
2. les nœuds situés sur une frontière du fluide autre que la membrane (entrée ou sortie du fluide),
3. les nœuds n'appartenant à aucune frontière.

Le programme fluide va travailler avec l'ensemble des nœuds. Le programme structure, lui, ne s'intéresse qu'aux nœuds de la membrane. Parmi ceux-ci, il doit également avoir accès aux propriétés physiques telle que les conditions aux frontières qu'il faut

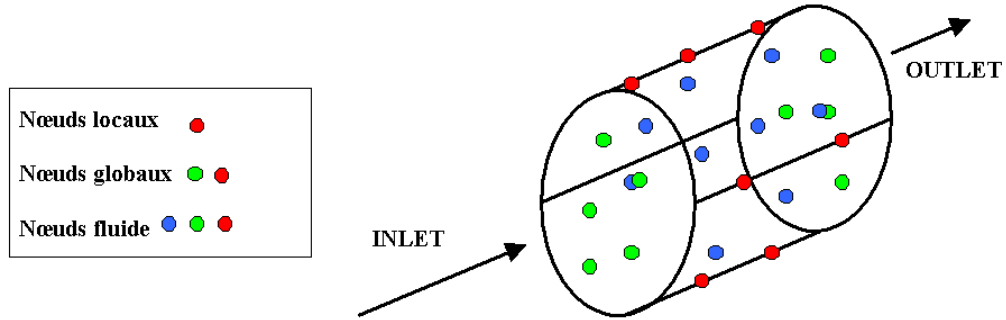


FIG. 6.3 – Illustration de la classification de nœuds.

appliquer. Par la suite, nous appellerons nœuds *globaux* l'ensemble des nœuds situés sur la frontière du fluide (catégories 1 et 2), et nœuds *locaux* ceux de la membrane (catégorie 1 uniquement).

Notons que les tétraèdres générés doivent être relativement réguliers dans le cas d'un couplage impliquant de grandes déformations, sans quoi l'écrasement d'un tétraèdre peut se produire lors du déplacement des nœuds. Dans ce cas, le programme fluide ne peut poursuivre la simulation.

2.2 Adaptation de la structure *mesh*

Il faut établir une méthode permettant au programme structure d'extraire les informations dont il a besoin dans le maillage global et d'établir une correspondance entre les nœuds globaux et locaux.

Pour cela, nous avons de nouveau recours aux tags. A la création du maillage, nous donnons un tag pouvant prendre quelques valeurs particulières à la membrane, et nous donnons d'autres valeurs aux tags des autres surfaces. Le programme *load_gmsh_2D.m*, chargé de lire le maillage produit par *Gmsh*, va alors ne prendre en considération que les éléments ayant un tag de membrane.

Les variables décrites dans le tableau 6.1 sont celles décrites auparavant, dans la section I.3.2.3.

Les quatre dernières variables du tableau contiennent les caractéristiques de tous les éléments de membrane. Parmi ces caractéristiques se trouvent les nœuds sur lesquels s'appuient ces éléments, dont les indices font directement référence à une ligne de *mesh.POS*. Le problème est que cette dernière variable contient les caractéristiques des nœuds globaux, et non des nœuds locaux. Or, notre programme est conçu pour travailler avec une liste des nœuds locaux seulement. Il faut donc une table de correspondance permettant de passer d'une liste ordonnée des nœuds locaux à une liste ordonnée des nœuds globaux, en l'occurrence *mesh.POS*.

mesh.nbNod	nombre de nœuds du maillage,
mesh.nbElm	nombre d'éléments du maillage,
mesh.nbBeams	nombre d'arêtes,
mesh.nbTriangles	nombre de triangles,
mesh.nbQuads	nombre de quadrangles,
mesh.POS	positions des nœuds du maillage
mesh.POINTS	description de tous les points particuliers,
mesh.BEAMS	description de toutes les arêtes,
mesh.TRIANGLES	description de tous les triangles,
mesh.QUADS	description de tous les quadrangles.

TAB. 6.1 – Variables préexistantes de la structure de données *mesh*.

Pour cela, nous avons adapté la structure de données *mesh*. Désormais, le programme *load_gmsh_2D.m*, va y ajouter trois nouvelles variables :

mesh.nbNodMembrane	nombre de nœuds locaux,
mesh.loc2glob	passage de l'indice local à l'indice global,
mesh.glob2loc	passage inverse.

Si sur la $i^{\text{ème}}$ ligne de *mesh.loc2glob* nous trouvons l'indice j , cela signifie que le nœud dont l'indice est i dans les variables locales est le nœud d'indice j dans les variables globales. La variable *mesh.glob2loc* réalise l'opération inverse. Lorsqu'à un nœud global ne correspond aucun nœud local, la valeur trouvée dans *mesh.glob2loc* est 0.

3 Communication fluide-structure

Comme expliqué plus haut, les deux programmes écrivent des fichiers avec leurs résultats et lisent les résultats l'un de l'autre.

3.1 Transmission des forces

Les résultats transmis par le programme fluide sont les forces appliquées par le fluide en chacun des nœuds de la membrane, sous la forme d'un fichier portant l'extension *.force*. La forme du fichier est la suivante :

Indice global du nœud	Coord. initiales			Forces		
	X	Y	Z	Fx	Fy	Fz
1
2
⋮	⋮	⋮	⋮	⋮	⋮	⋮
Nombre de nœuds globaux

Le fichier contient les forces appliquées sur les nœuds globaux. Il faut donc utiliser l'une des deux variables de correspondance pour en extraire l'information utile.

Le programme *readP.m* a pour rôle de lire le fichier *.force*. Ils stocke d'abord toutes les forces dans une variable temporaire. Puis, en parcourant la variable *mesh.loc2glob*, une boucle lit l'indice global correspondant à chaque nœud et trouve les forces correspondant aux nœuds en lisant la variable temporaire. Il écrit ainsi une variable *Forces* reprenant toutes les forces appliquées aux nœuds locaux.

3.2 Transmission des déplacements

Lorsque le code structure termine un calcul, il écrit un fichier texte contenant les déformations calculées (par rapport à la configuration de référence). Le programme *writeOut.m* effectue l'opération. Le fichier a la forme suivante :

Indice global du nœud	Déplacement		
	X	Y	Z
1	.	.	.
2	.	.	.
⋮	⋮	⋮	⋮
Nombre de nœuds locaux	.	.	.

Remarquons que la première colonne contient l'indice *global* des nœuds, puisque le programme fluide ne connaît pas les variables de correspondance de *mesh*. De cette manière, les deux codes restent totalement indépendants.

Deux entrées sont ajoutées aux inputs du programme structure. Il s'agit des chemins d'accès du fichier *.force* et du fichier des déplacements.

4 Prise en compte de forces extérieures quelconques

Globalement, notre code n'a pas beaucoup changé lorsque nous l'avons adapté au couplage. Nous savions dès le début quel en serait l'usage et nous l'avons conçu dans cette optique. Néanmoins, quelques améliorations ont été faites. Outre la modification de *load_gmsh_2D.m* présentée ci-dessus, et l'ajout d'inputs et d'outputs, nous avons modifié le code au niveau du calcul des forces externes afin de l'adapter aux forces transmises par le programme fluide. En effet, celles-ci ne sont pas constantes le long de la membrane et ne lui sont pas nécessairement perpendiculaires.

Forces non-constantes

Dans la section I.2.2.3, nous avons fourni l'expression discrète des forces externes pour une pression p constante sur toute la membrane (équation (2.31)) :

$$\mathbf{F}_{\text{ext}}^e = \int_{\Omega} p \mathbf{N}^T \cdot (\mathbf{g}_1 \wedge \mathbf{g}_2) ds_1 ds_2. \quad (6.1)$$

Pour prendre en compte des pressions variables, nous utilisons l'expression suivante des forces de pression :

$$p(s_1, s_2) = \mathbf{N}(s_1, s_2) \cdot \mathbf{P}, \quad (6.2)$$

ou en développant complètement l'écriture pour un élément triangulaire à trois nœuds :

$$p(s_1, s_2) = \begin{bmatrix} \phi_1(s_1, s_2) & \phi_2(s_1, s_2) & \phi_3(s_1, s_2) \end{bmatrix} \cdot \begin{bmatrix} P_1 \\ P_2 \\ P_3 \end{bmatrix}, \quad (6.3)$$

où P_1, P_2, P_3 sont les pressions aux nœuds de l'élément.

Pour un élément quadrangulaire à quatre nœuds, cela donne :

$$p(s_1, s_2) = \begin{bmatrix} \phi_1(s_1, s_2) & \phi_2(s_1, s_2) & \phi_3(s_1, s_2) & \phi_4(s_1, s_2) \end{bmatrix} \cdot \begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \end{bmatrix}, \quad (6.4)$$

où P_1, P_2, P_3, P_4 sont les pressions aux nœuds de l'élément.

Forces de directions quelconques

Pour des forces non-constantes et non-perpendiculaires à la membrane, nous devons réécrire le terme du travail virtuel des forces extérieures pour un élément e dans le Principe des Travaux Virtuels (équation 2.14) :

$$\delta w_{\text{ext}}^e = \int_{S^e} \mathbf{f} \cdot \delta \mathbf{u}^T dS. \quad (6.5)$$

Le terme \mathbf{f} est une densité de forces, tout comme $p \mathbf{n}$ dans l'expression développée précédemment. En prenant en compte des densités de force non-constantes, \mathbf{f} s'écrit :

$$\mathbf{f} = \mathbf{N} \cdot (\mathbf{k} : \mathbf{F}), \quad (6.6)$$

où \mathbf{F} est une matrice $3n^e \times 1$ contenant les trois composantes des forces extérieures calculées aux différents nœuds de l'élément, et \mathbf{k} est une matrice de facteurs de proportionnalité permettant de transformer les forces \mathbf{F} aux nœuds en une distribution de forces sur les différents éléments adjacents. Pour des triangles, il s'écrit :

$$\begin{bmatrix} k_1 & k_1 & k_1 & k_2 & k_2 & k_2 & k_3 & k_3 & k_3 \end{bmatrix}^T. \quad (6.7)$$

L'utilisation de ce rapport est une méthode permettant de passer d'une force ponctuelle en un nœud à une densité de force s'appliquant autour du nœud. La figure 6.4 permet d'illustrer ce rapport. Nous considérons que la force ponctuelle se répartit sur la zone grisée. Celle-ci a pour aire la somme des aires de tous les éléments adjacents divisée par le nombre de nœuds par élément.

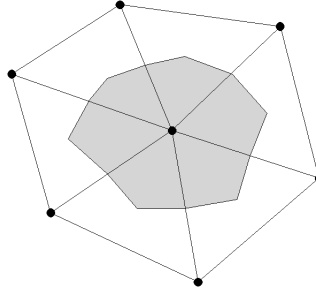


FIG. 6.4 – Aire de répartition d'une force nodale.

Les deux autres facteurs $\delta \mathbf{u}^T$ et dS se développent comme auparavant :

$$\delta \mathbf{u}^T = \delta \mathbf{U}^{\mathbf{e}T} \cdot \mathbf{N}^T, \quad (6.8)$$

$$dS = \| \mathbf{g}_1 \wedge \mathbf{g}_2 \| ds_1 ds_2. \quad (6.9)$$

Le travail des forces extérieures s'écrit donc finalement :

$$\delta \omega_{ext}^e = \delta \mathbf{U}^{\mathbf{e}T} \cdot \int_{\Omega} \mathbf{N}^T \cdot \mathbf{N} \cdot (\mathbf{k} : \mathbf{F}) \| \mathbf{g}_1 \wedge \mathbf{g}_2 \| ds_1 ds_2, \quad (6.10)$$

ce qui nous donne l'expression suivante des forces extérieures aux nœuds :

$$\mathbf{F}_{ext}^e = \int_{\Omega} \mathbf{N}^T \cdot \mathbf{N} \cdot (\mathbf{k} : \mathbf{F}) \| \mathbf{g}_1 \wedge \mathbf{g}_2 \| ds_1 ds_2. \quad (6.11)$$

Remarquons qu'il est maintenant nécessaire d'intégrer une fonction quadratique pour les triangles, et donc d'utiliser une méthode d'intégration plus élevée.

Chapitre 7

Applications

La vérification d'un programme est une étape importante qu'il ne faut pas négliger. Pour ce faire, la théorie peut parfois nous être très utile, comme pour les vérifications faites sur le code membrane. Néanmoins, il existe peu d'études théoriques sur l'interaction fluide-structure, et la discipline ne s'intéresse aux membranes que depuis peu de temps. Par ailleurs, peu de résultats expérimentaux sont disponibles, la plupart d'entre eux restant confinés dans le milieu industriel. La réalisation d'expériences de couplage serait donc une entreprise intéressante qui mériterait que du temps lui soit consacré.

Dans ce chapitre, nous nous attachons principalement à montrer quelques applications pratiques que notre programme peut modéliser. Les premières sont plus théoriques et nous permettent de faire des vérifications sur les résultats. Les suivantes sont des simulations plus concrètes.

1 Ecoulement laminaire non-visqueux en conduite

Dans cette section, nous observons ce qui se passe dans le cas d'un couplage simple, celui d'une conduite membranaire à l'intérieur de laquelle un écoulement non-visqueux se produit.

Dispositif et paramètres

Le maillage des parois du dispositif est représenté à la figure 7.1. Il est composé d'une conduite fixe d'entrée (en bleu), d'une conduite de sortie (bleu clair), et d'une membrane cylindrique au centre (vert). Le maillage 3D s'appuie sur celui-ci. Les conduites fixes sont destinées à encaisser les perturbations éventuelles pouvant survenir aux frontières d'entrée et de sortie du fluide.

Les valeurs suivantes sont prises pour le rayon des trois objets R , la longueur des conduites fixes L_{in} et L_{out} , la longueur caractéristique des mailles l_c , la masse

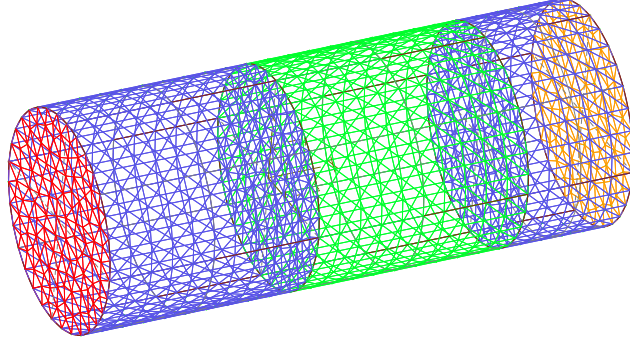


FIG. 7.1 – Maillage des parois utilisé pour les tests sur membrane cylindrique.

volumique du fluide ρ , sa viscosité ν , sa vitesse moyenne \bar{V} , la différence de pression entre l'intérieur et l'extérieur à l'entrée P_{in} , le nombre de Mach, l'épaisseur initiale de la membrane H , ses coefficients de Mooney-Rivlin C_1 et C_2 , sa longueur L_m , sa masse volumique ρ_m et sa constante d'amortissement C_{amort} :

R	1	m
L_{in}	2	m
L_{out}	1	m
l_c	0.2	m
ρ	1000	kg/m^3
ν	$2 \cdot 10^{-5}$	m^2/s
\bar{V}	0.08	m/s
P_{in}	500	Pa
Ma	0.1	—
H	10	mm
C_1	100 000	Pa
C_2	10 000	Pa
L_m	2	m
ρ_m	1500	kg/m^3
C_{amort}	50	kg/s

Par abus de langage, nous désignerons la différence de pression entre les deux côtés de la paroi des conduites par *pression interne* ou simplement *pression*.

Les équations résolues sont les équations d'Euler. Le fluide est donc considéré comme non-visqueux.

Une remarque importante doit être faite quant à la nature du fluide étudié. Malgré la densité de 1000 kg/m^3 , pour le programme le fluide est autre que de l'eau. En effet, les équations résolues par le code fluide sont adimensionnalisées, et nous devons donc nous intéresser aux nombres adimensionnels que le code va prendre en compte. Il se

peut donc que le fluide virtuel que nous étudions ne corresponde pas à un fluide connu en particulier.

La démarche à utiliser pour introduire un problème sera donc de trouver les paramètres adimensionnels de l'écoulement à partir des paramètres dimensionnels connus, et de restituer au code fluide un jeu de paramètres aboutissant aux mêmes nombres adimensionnels. Ceux-ci sont le nombre de Reynolds et le nombre de Mach.

Dans ce cas-ci, l'écoulement que nous allons observer possède Re de 4000 (calculé via l'équation 5.8) et un nombre de Mach de 0,1.

Résultats

Commençons par observer ce qui se passe dans les premières fractions de secondes. La figure 7.2 montre respectivement les pressions et les vitesses du fluide après 0.05 secondes.

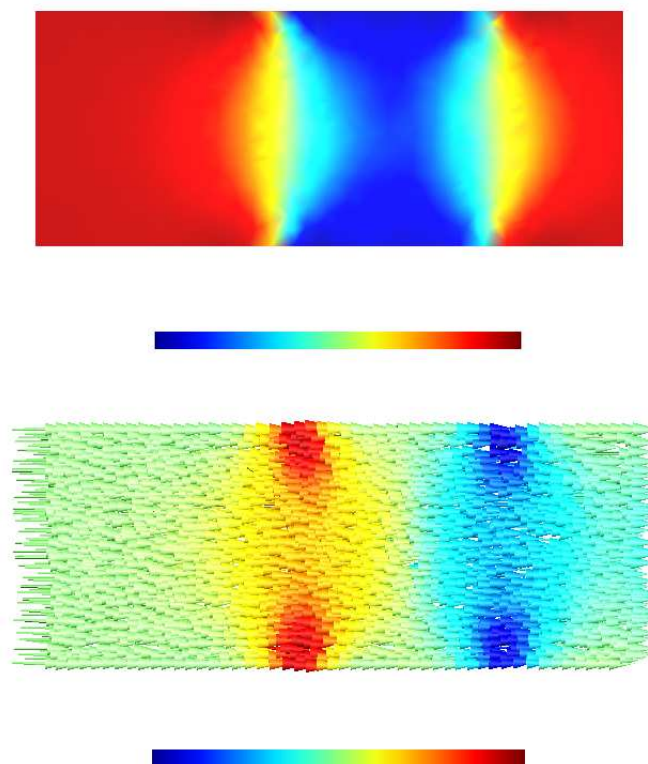


FIG. 7.2 – Pressions et vitesses du fluide en $t = 0.05$ s.

La zone bleue sur la figure des pressions est une zone de dépression due à l'extension de la membrane. Celle-ci a des conséquences sur la distribution des vitesses. Les graphes

représentés sur la figure 7.3 montrent l'augmentation rapide du rayon de la section médiane de la membrane, ainsi que la dépression qui s'ensuit. La pression est relevée en un point situé sur la section médiane de la membrane. Nous voyons également que

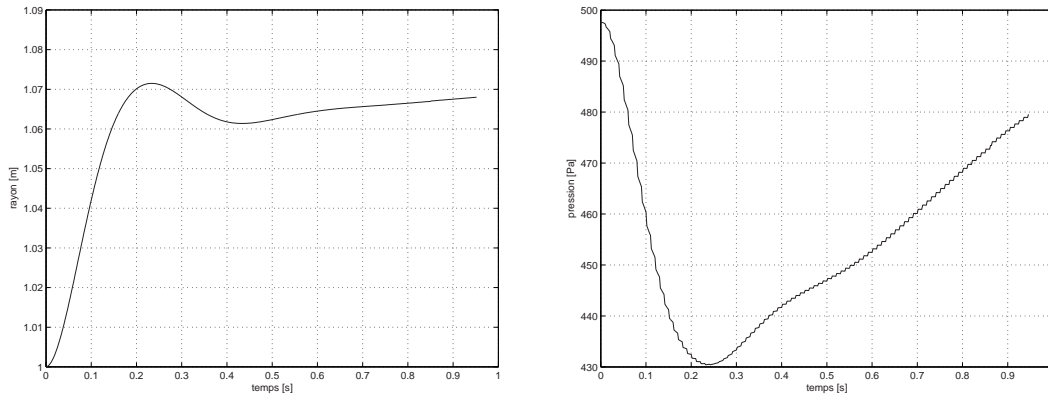


FIG. 7.3 – Rayon maximum de la membrane et pression du fluide au cours des premières secondes.

le léger retour de la membrane a pour effet d'accentuer brièvement la remontée de la pression.

Par la suite, la dépression se propage vers les extrémités du dispositif. La figure 7.4 montre les pressions toutes les 0.2 secondes jusqu'à $t = 2$ secondes. Nous voyons l'onde se propager à vitesse plus élevée vers la sortie que vers l'entrée. En effet, une onde de pression dans un fluide au repos se propage à la vitesse du son c , mais si le fluide est en mouvement, elle se propagera à la vitesse $c + V$ dans le sens du courant, et $c - V$ à contre-courant si le fluide est en mouvement.

Examinons à présent la déformation de la membrane. La figure 7.5 montre les variations du rayon et de la pression au point étudié et la figure 7.6 la membrane déformée.

L'onde de pression dont nous avons parlé ci-dessus continue à déplacer et se répercute aux frontières du domaine. Les oscillations ne sont pas amorties car il n'y a pas de terme visqueux dans les équations d'Euler. Nous pouvons noter que les pressions et le rayon ont des évolutions fort similaires. La pression moyenne est d'environ 496.6 Pa , et le rayon moyen de $1,071 \text{ m}$. En calculant la déformée de la même membrane pour une pression uniforme de 500 Pa (sans couplage), le rayon de la section considérée est de $1,072 \text{ m}$.

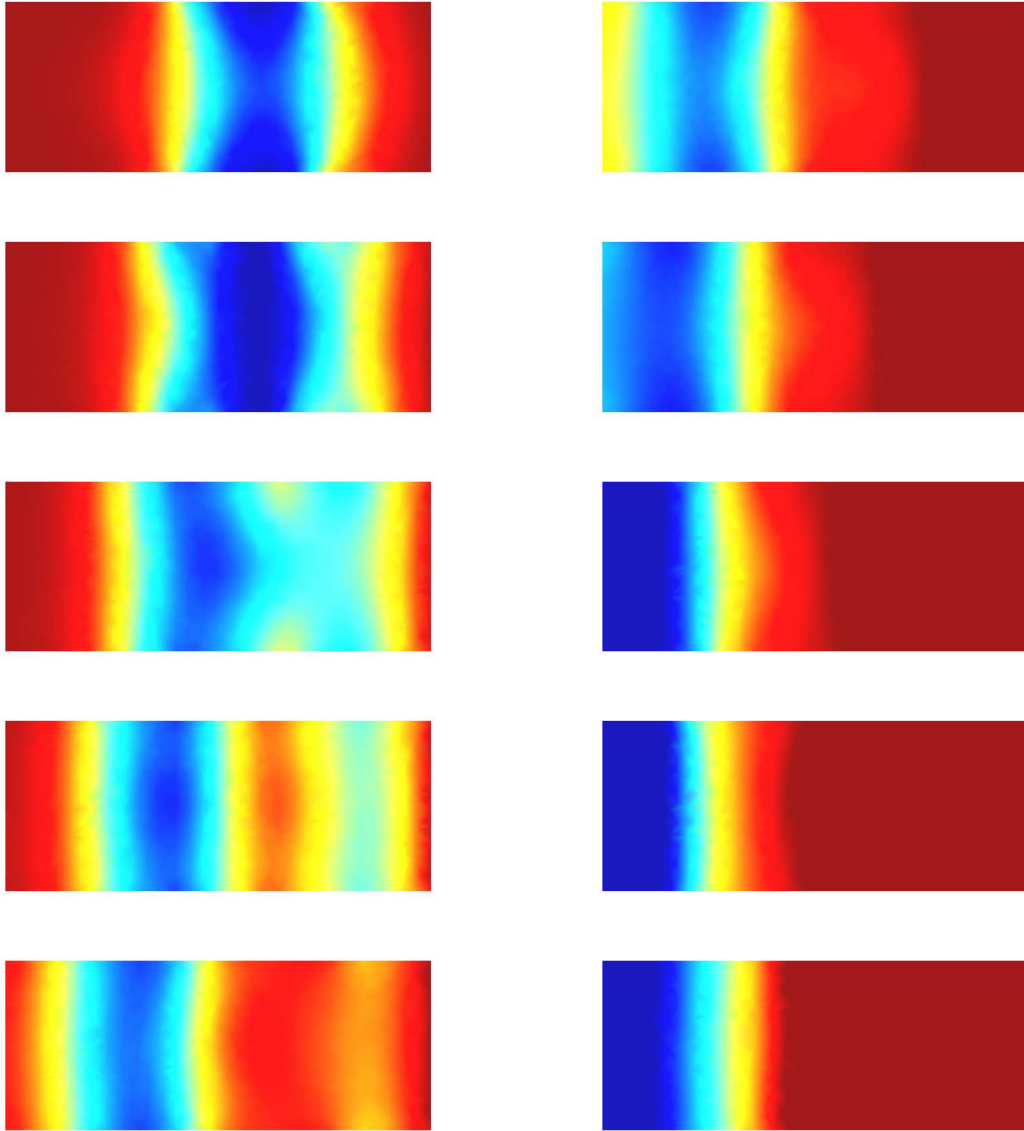


FIG. 7.4 – Relevé de la pression à l'intérieure de la conduite toutes les 0.2 secondes entre le temps 0 et le temps 2 s.

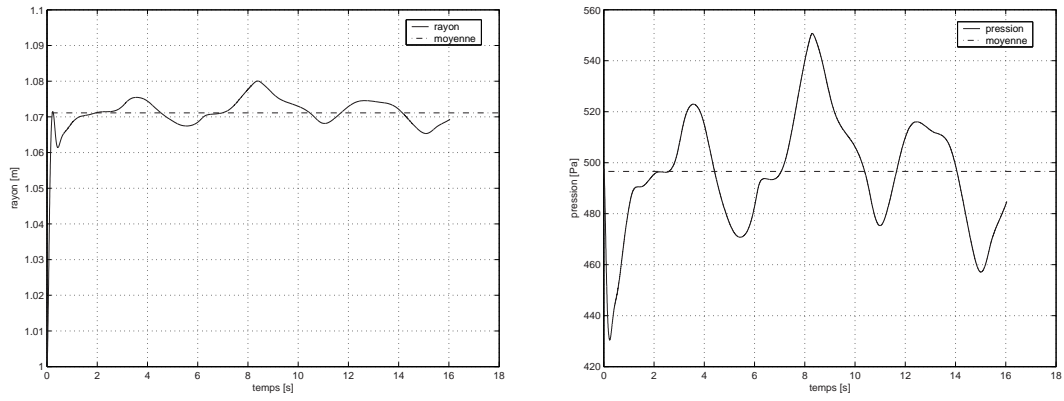


FIG. 7.5 – Evolution du rayon et de la pression sur la section médiane de la membrane.

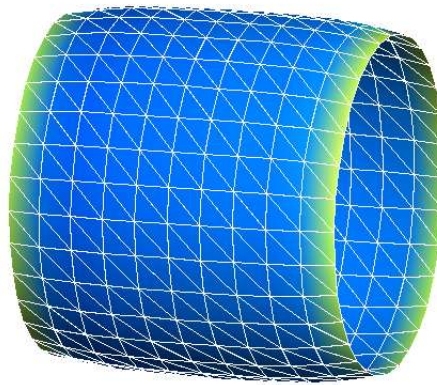


FIG. 7.6 – Déformation de la membrane cylindrique.

2 Écoulement de Poiseuille entre deux plans

L'écoulement de Poiseuille est un écoulement dont la viscosité est prépondérante sur la convection. Si nous considérons un écoulement entre deux plans distants de 1 m , avec $Re = 100$, nous savons que le profil des vitesses a la forme présentée à la figure 7.7, et que la vitesse maximum V_{max} sur un profil vaut 1,5 fois la vitesse moyenne \bar{V} .

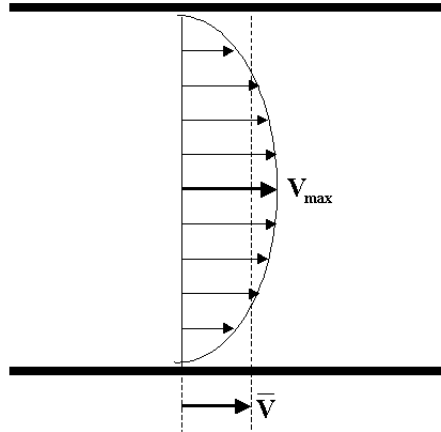


FIG. 7.7 – Profil de vitesses pour un écoulement de Poiseuille entre deux plans caractérisé par un nombre de Reynolds de 100.

Nous simulons cet écoulement entre deux membranes. Les plans fixes sont simplement remplacés par des membranes et les paramètres du fluide sont imposés de telle sorte que $Re = 100$. Le débit est de $93\text{ m}^3/\text{s}$ (écoulement fictif).

Une zone fixe est prévue en amont afin d'absorber les perturbations dues aux changements de distributions de vitesses. En effet, nous faisons entrer le fluide à une vitesse constante sur toute la section, alors que la distribution des vitesses doit adopter une forme parabolique. L'évolution des vitesses à l'entrée est montrée à la figure 7.8. Les faibles vitesses près des parois provoquent une déviation des flux vers le centre de la section. L'écoulement prend rapidement la forme voulue.

Une fois la membrane au repos, cinq profils de vitesses ont été relevés à intervalles réguliers (0, 25, 50, 75, 100 % de la longueur) au droit de la membrane. Ces profils ainsi que la déformation sont représentés sur la figure 7.9. Les vitesses sont représentées sur la géométrie initiale.

La vitesse maximale du premier profil est de 140 m/s , alors que la vitesse moyenne vaut, en prenant une largeur unitaire du volume étudié :

$$V = \frac{Q}{A} = \frac{93\text{ m}^3/\text{s}}{1\text{ m} \cdot 1\text{ m}} = 93\text{ m/s}. \quad (7.1)$$

Nous vérifions donc bien la relation sur la vitesse maximale puisque $1,5 \cdot 93\text{ m/s} = 139,5\text{ m/s} \simeq 140\text{ m/s}$. Les vitesses maximales des autres profils sont respectivement : 130, 120, 122 et 137 m/s , alors que la déformation de la membrane nous

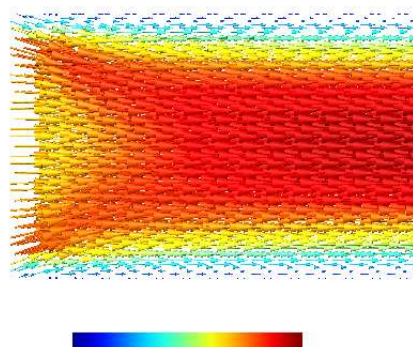


FIG. 7.8 – Vitesses à l'entrée de la section pour un écoulement de Poiseuille.

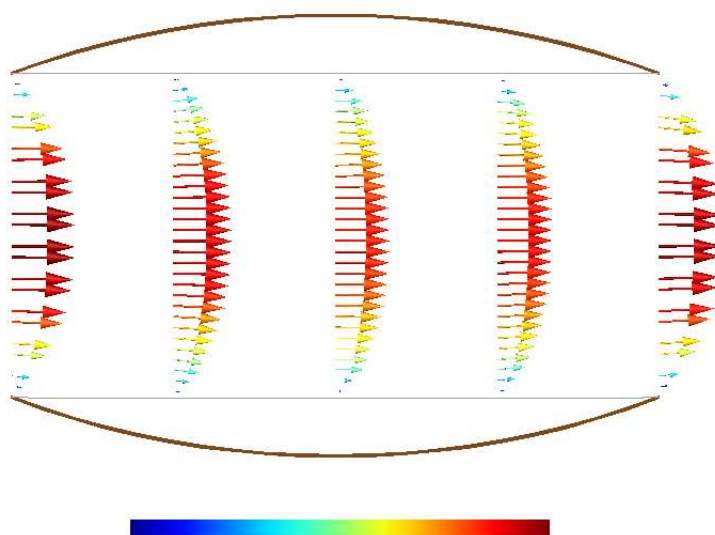


FIG. 7.9 – Profils de vitesses entre les deux membranes et déformation des parois ; les vitesses sont représentées sur la géométrie initiale.

permet de calculer les sections (1.161, 1.214, 1.163 et 1 m^2), les vitesses moyennes (80.1, 76.6, 80.0 et 93 m/s), et d'en déduire les vitesses maximales attendues suivantes : 120.2, 114.9, 120.0 et 139.5 m/s . Les différences entre vitesses maximales relevées et attendues sont donc : 9.8, 5.1, 2.0 et -2.5 m/s .

Les différences importantes dans les premiers profils peuvent trouver une explication dans la direction de l'écoulement et la grande vitesse de celui-ci. En effet, l'élargissement est relativement brusque et les profils de vitesses ne peuvent s'adapter suffisamment vite. Lorsque le fluide a franchi la moitié de la longueur, le profil est pratiquement celui attendu. Un nouvel espace d'adaptation est nécessaire pour qu'il retrouve sa forme pour une hauteur de 1 m , ce qui explique le signe négatif de la

dernière valeur.

Par ailleurs, nous relevons une pression intérieure quasi-constante de 53 Pa le long de la membrane, ce qui donne un déplacement nodal maximal de 0.107 m avec le programme structure seulement, soit exactement la déformation de la membrane obtenue via le couplage. Notons que les forces visqueuses et le mouvement faiblement divergeant du fluide au centre du dispositif font que la membrane est légèrement plus déformée en aval qu'en amont.

3 Application médicale

Le domaine médical travaille énormément avec des membranes. Le corps humain en contient beaucoup et les prothèses qui les remplacent sont aussi des membranes. Certains outils manipulés par les chirurgiens en contiennent également. Une des applications importantes concernées par la théorie des membranes est la simulation de l'interaction fluide-structure sur les aortes. Celles-ci sont soumises à de nombreux cycles de pressions et peuvent parfois souffrir de déformations particulières, par exemple dans le cas d'un anévrisme. L'anévrisme artériel se définit par une perte du parallélisme des bords d'une artère, entraînant une dilatation localisée permanente de l'artère avec une augmentation de diamètre d'au moins 50 % par rapport au diamètre normal théorique de l'artère considérée. Pour remédier à ce problème on pratique la greffe d'anévrisme de l'aorte abdominale sous-rénale par prothèse aorto-bi-iliaque [31], c'est à dire, une ablation de l'anévrisme et le placement d'une prothèse aortique. L'anévrisme de l'aorte abdominale sous-rénale (A.A.A.) se situe entre l'origine des artères rénales en haut et la bifurcation aorto-iliaque en bas, sur le segment en V de l'aorte.

Simulation numérique

Nous modélisons cette prothèse aortique au niveau de la bifurcation de l'aorte. Nous pouvons, de manière simplifiée, remplacer l'aorte par une membrane tubulaire. Le programme peut calculer les déformations de la membrane en fonction de la pression mais aussi l'écoulement intérieur. Les caractéristiques des aortes sont bien connues et les différents coefficients nécessaires pour la simulation sont facilement accessibles.

La déformée que nous avons obtenue par couplage est représentée à la figure 7.13. Les couleurs représentent les normes des forces internes aux nœuds. Nous pouvons constater que les parois des deux petits vaisseaux subissent une contrainte plus élevée, mais que celle-ci est variable à cause des transitoires du fluide.

La simulation effectuée est simplifiée par rapport à un cas réel et ne tient pas compte de nombreux facteurs, tels que la réaction de la matière environnante de l'aorte, les pulsations du cœur, etc. Nous voyons tout de même qu'il est possible d'obtenir des résultats relativement complexes via des simulations numériques, la seule limite étant le temps disponible pour les réaliser.

Perfectionnements et apports au programme

Afin d'avoir plus de détails sur la réalité des phénomènes, les pressions et les vitesses à l'entrée du modèle numérique de l'aorte devraient être variables dans le temps. Les pressions au cours du temps à l'intérieur de celle-ci devraient adopter la forme présentée à la figure 7.14.

L'imagerie médicale peut nous fournir les géométries, voire directement les maillages nécessaires, lorsque des formes plus compliquées apparaissent, comme des anévrismes.

L'artère aorte et ses branches

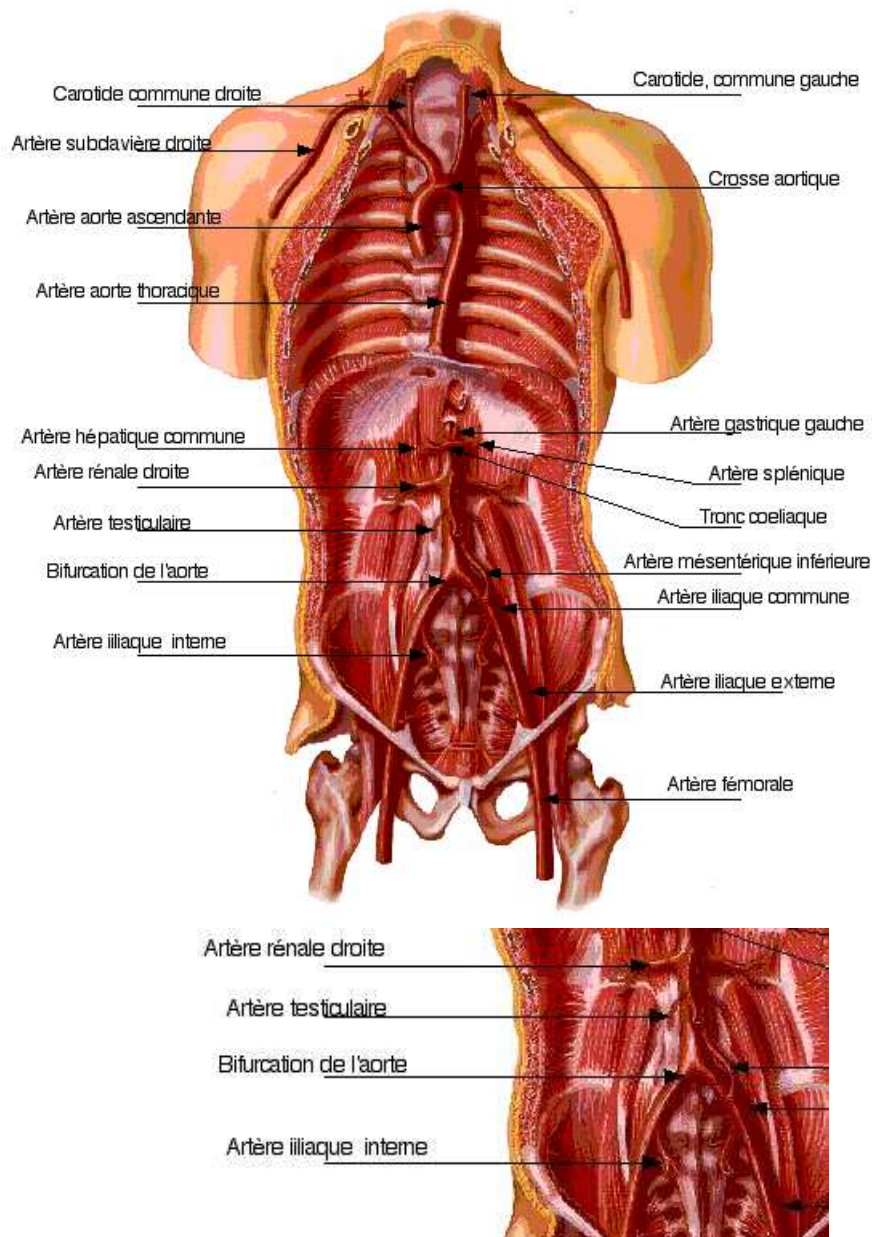


FIG. 7.10 – Disposition d'une aorte.

Le programme permettrait de mieux comprendre les phénomènes intervenant dans le corps d'un patient, d'avoir une meilleure visualisation d'ensemble, ou de mieux préparer les prothèses et ainsi conforter et assister les décisions des médecins.

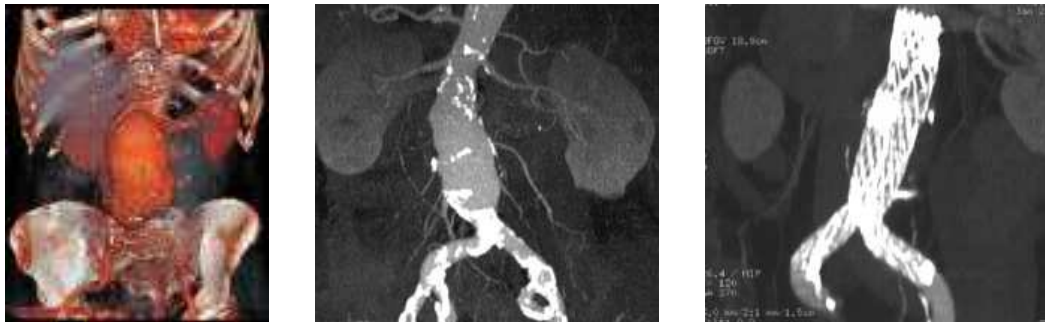


FIG. 7.11 – Traitement d'un anévrisme de l'aorte abdominale sous-rénale et pose d'une prothèse aortique [33].



FIG. 7.12 – Prothèses aortiques [31].

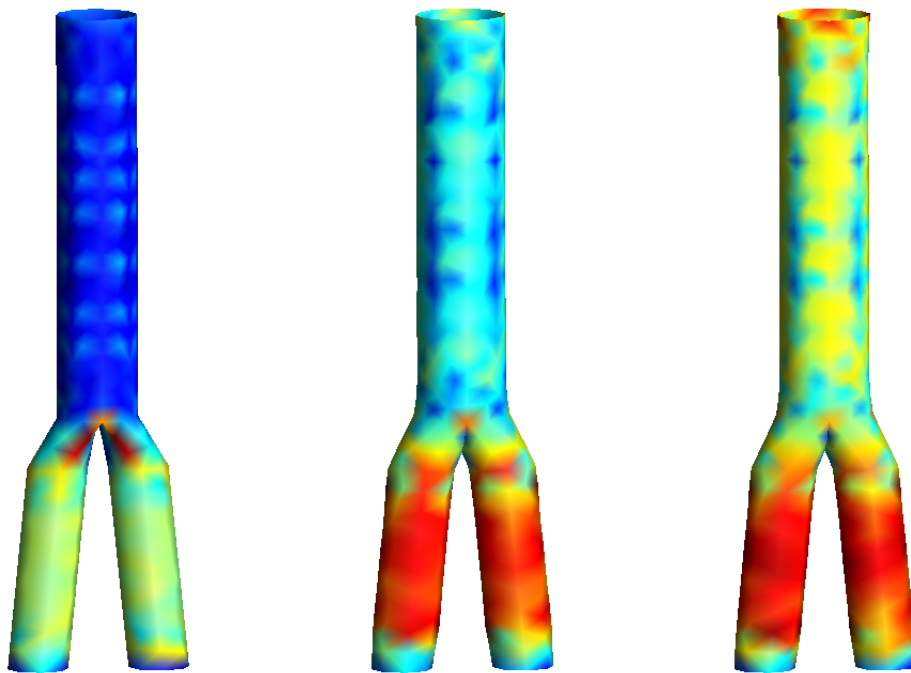


FIG. 7.13 – Modélisation d'une aorte en interaction fluide-structure.

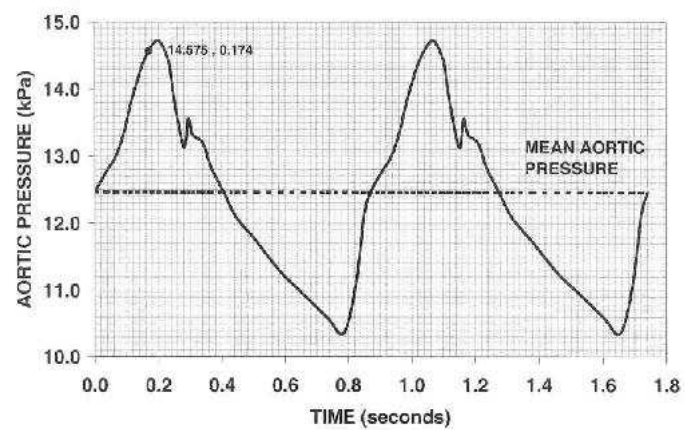


FIG. 7.14 – Pression mesurée à l'intérieur d'une aorte.

4 Descente d'un parachute

Le comportement d'un parachute descendant vers le sol est intéressant tant du point de vue structure que fluide. Le couplage des codes permet de calculer la portance d'un parachute, d'optimiser sa forme et de connaître les phénomènes se produisant dans son sillage.

Nous étudions en guise d'exemple une membrane rectangulaire de 4 mètres de large et de 6,4 mètres de long, dont les quatre coins sont maintenus dans un même plan, et soumise à un écoulement identique à celui se produisant autour d'un parachute :

ρ_{air}	1,2	kg/m^3
L_{ref}	5	m
c	300	m/s
V	5	m/s
ν	$1.45 \cdot 10^{-5}$	m^2/s

où c est la vitesse du son. Avec ces valeurs, l'écoulement doit être caractérisé par les grandeurs adimensionnelles suivantes :

$$Re = \frac{L_{ref} V}{\nu} = 1.7 \cdot 10^6, \quad (7.2)$$

$$Ma = \frac{V}{c} = 0.016. \quad (7.3)$$

En ce qui concerne la membrane, nous prenons les caractéristiques suivantes :

C_1	100 000	Pa
C_2	0	Pa
H	1	mm
ρ_m	1500	kg/m^3
C_{amort}	20	kg/s

Le maillage employé est représenté à la figure 7.15.

Résultat

Les quatre coins du parachute sont libres de se mouvoir dans le plan initial de la membrane, et sont retenus dans la direction de l'écoulement. On pourrait penser qu'avec ces conditions aux frontières, le parachute devrait se replier sur lui-même. Au vu des résultats, il n'en est rien. Il se déploie et trouve une situation stable. La figure 7.16 montre quelques positions occupées par le parachute durant sa déformation. La dernière image montre la situation dans laquelle il ne bouge plus.

Les points d'accroche des cordes se rapprochent les uns des autres au cours du déploiement. L'aire de la surface projetée du parachute diminue, ce qui réduit la portance. La figure 7.17 montre le parachute vu du dessus.

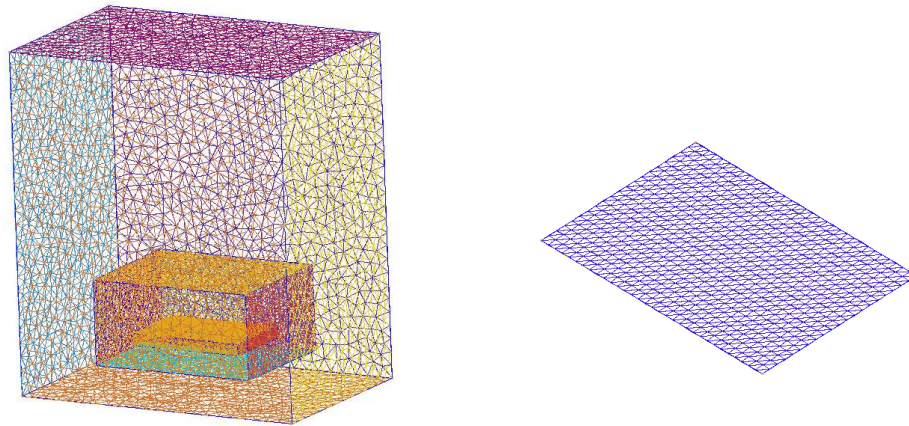


FIG. 7.15 – Maillages des frontières du fluide et de la structure, utilisés pour modéliser un parachute rectangulaire.

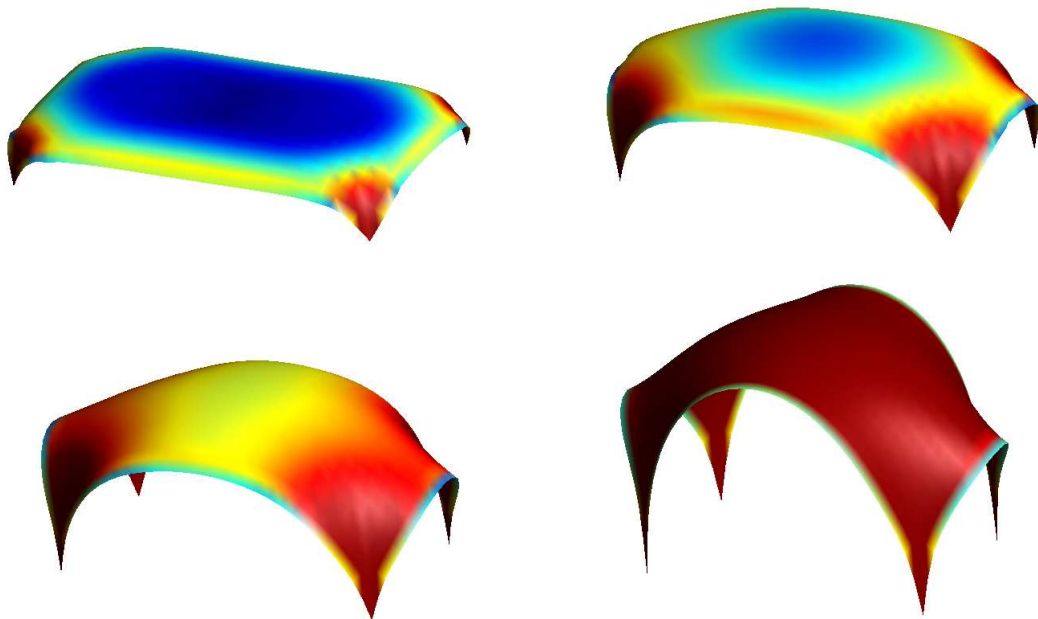


FIG. 7.16 – Déformations du parachute à quatre instants différents.

La figure 7.18 montre les champs des vitesses et des pressions. Les données sont représentées autour de la configuration initiale du parachute. Nous pouvons observer

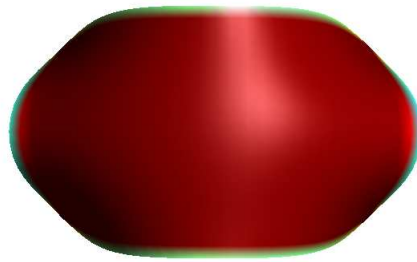


FIG. 7.17 – Vue du dessus de la déformée finale du parachute.

une zone de recirculation d'environ deux mètres de hauteur au-dessus du parachute. Le sillage des vitesses est beaucoup plus long : l'influence de l'obstacle dans l'écoulement dépasse la longueur représentée. Au niveau des pressions, la surpression sous le parachute est manifeste, de même que la dépression par dessus. La différence entre les deux faces du parachute est en moyenne de 20 Pa . Avec une surface projetée de 22 m^2 environ, la force portante est approximativement de 440 N , de quoi porter une charge de 45 kg . Un homme de poids moyen descendant à cette vitesse (5 m/s) va donc accélérer jusqu'à une vitesse de 7 ou 8 mètres par seconde avant que la différence de pression ne soit suffisante.

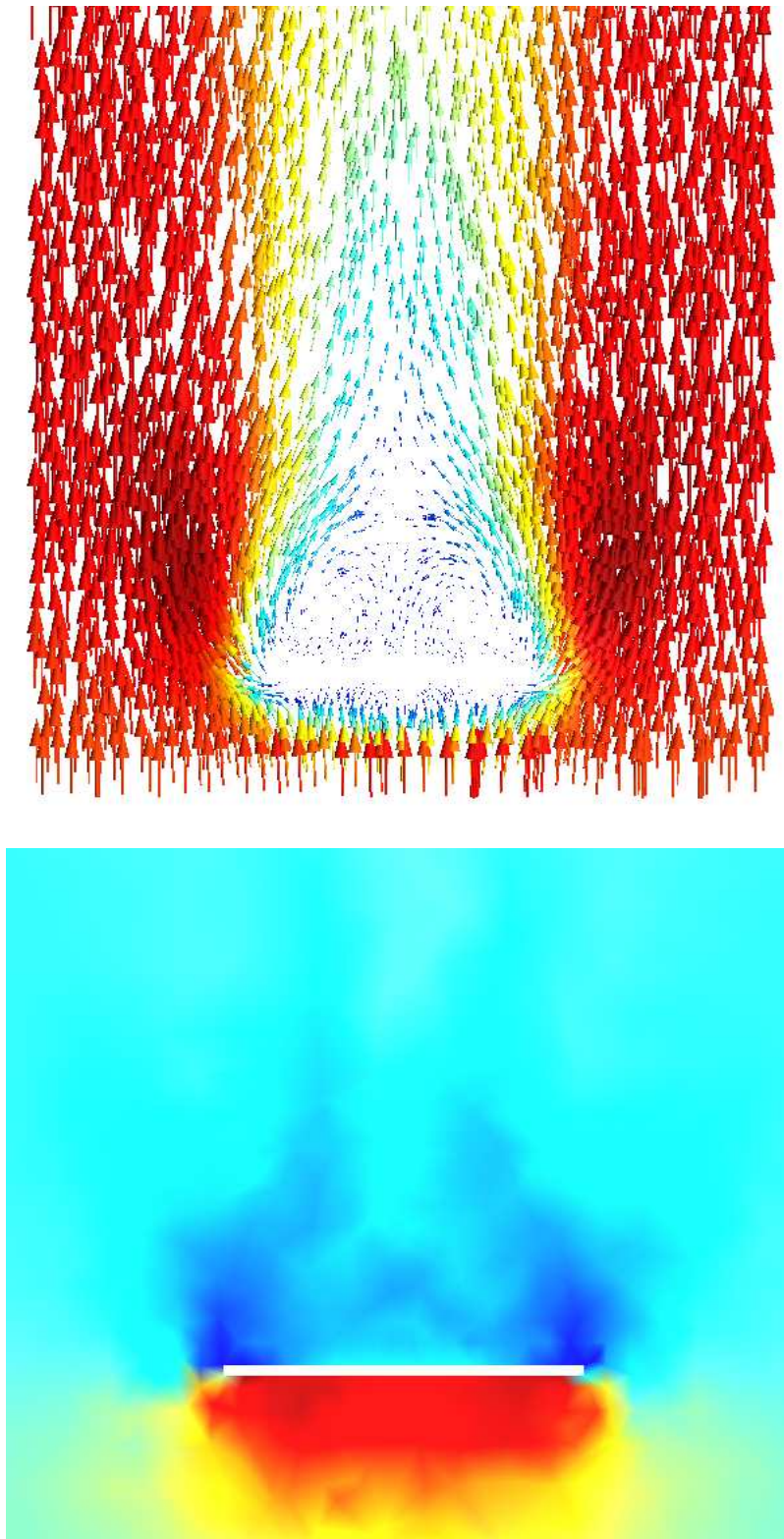


FIG. 7.18 – Champs des vitesses et des pressions représentés en coupe au centre du parachute dans l'une des deux directions principales. Les valeurs sont représentées sur la géométrie de la configuration initiale.

5 Gonflement d'un airbag

Depuis quelques années, les airbags ont fait leur apparition dans l'automobile. La technologie est récente et en plein développement. D'autres domaines où le souci de sécurité est croissant pourraient faire appel à des systèmes équivalents. Pour modéliser un airbag de manière réaliste, il est nécessaire de faire appel à un programme de couplage fluide-structure afin de pouvoir prendre en compte l'interaction survenant lors du gonflement.

Nous avons effectué la modélisation du gonflement d'un airbag en prenant pour point de départ une double paroi cylindrique mise « à plat ». Une large ouverture sur la paroi arrière permet l'entrée de l'air.

La figure 7.20 montre l'airbag dans plusieurs positions. Son gonflement est représenté par la figure 7.19.

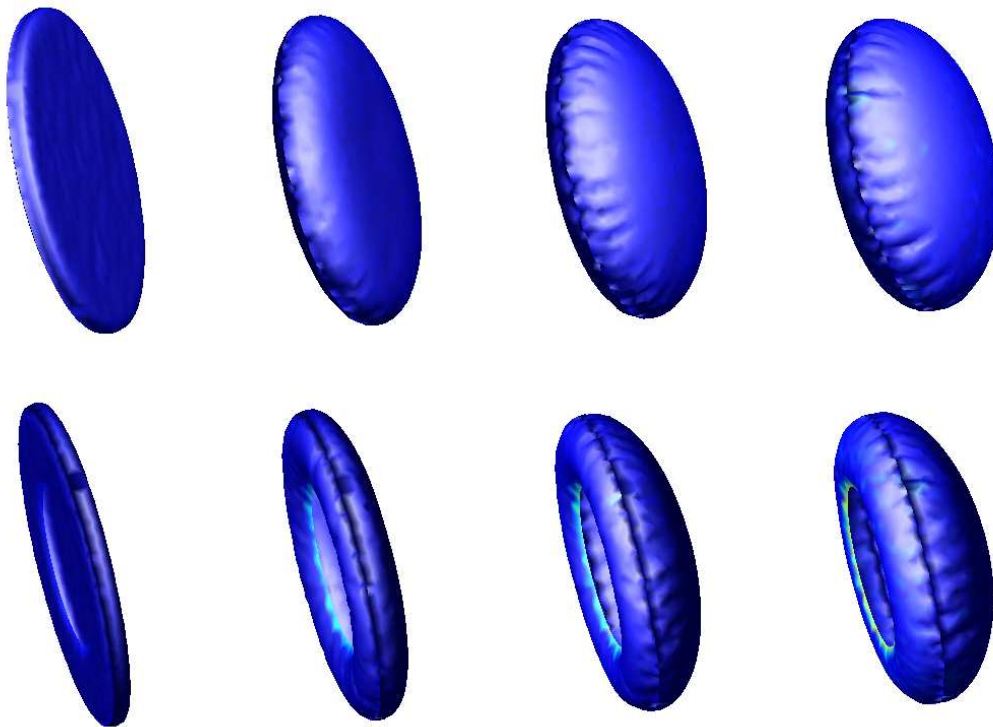


FIG. 7.19 – Déformations successives obtenues par couplage fluide-structure pour la modélisation d'un airbag.

Nous pouvons remarquer que le tissu se froisse le long des bords. Il s'agit de plis survenant aux frontières des éléments. Nous constatons donc qu'avec un maillage suffisamment fin, le froissement est bien représenté et fournit une déformation réaliste.

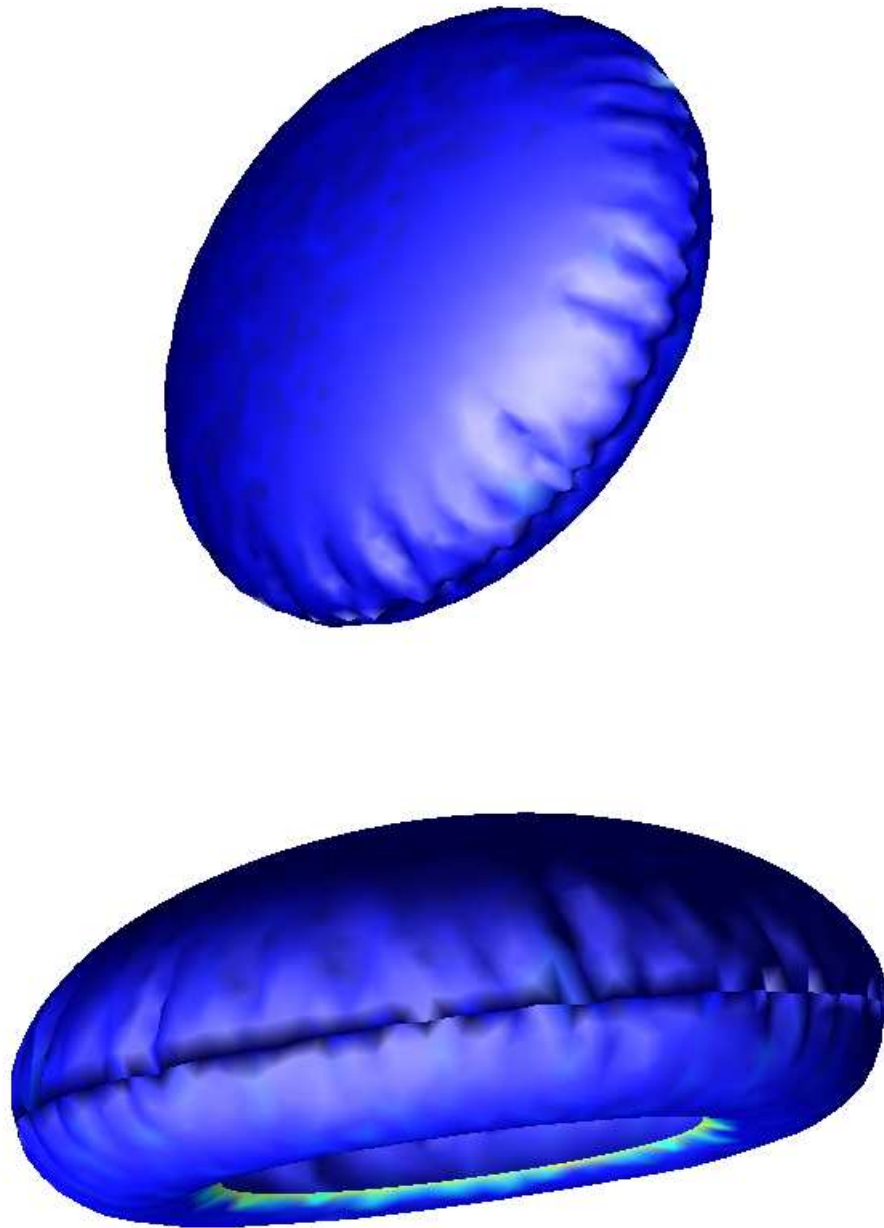


FIG. 7.20 – Déformations obtenues par couplage fluide-structure pour la modélisation d'un airbag.

Avec cet exemple se terminent les applications que nous avons parcourues. Les cas étudiés nous montrent que le domaine d'étude de l'interaction fluide-structure est vaste, et que le calcul numérique peut y jouer un rôle important sans pour autant demander de lourds investissements.

Conclusion

Objectifs atteints

Les objectifs de ce mémoire, à savoir la création d'un code membrane tridimensionnel et le couplage avec un programme fluide, ont été atteints. Le premier objectif a nécessité un grand plongeon dans une théorie qui nous était inconnue jusque là : la théorie des grandes déformations. Un regard critique sur nos résultats et beaucoup de patience nous ont permis de l'appliquer en implémentant un programme exact et cohérent, comme le montrent les vérifications réalisées.

Le couplage avec un programme fluide était un objectif ambitieux, dans la mesure où la réalisation du code membrane était en soi un travail difficile. Le couplage ne pouvait se faire sans entrer dans le domaine complexe de la mécanique des fluides. Nous avons dû pour ce faire nous familiariser avec la résolution de problèmes adimensionnels et tenter de mettre de l'ordre dans la multitude de paramètres qu'impliquait un tel couplage.

Plusieurs exemples d'interactions ont pu être testés. Nous avons montré que des applications pratiques aussi diverses que l'étude de parachutes, d'airbags ou encore de vaisseaux sanguins peuvent être modélisées via un code d'interaction fluide-membrane. Nous avons également pu faire quelques vérifications quant aux valeurs numériques fournies.

Pistes de réflexion

Tout au long de la réalisation de ce mémoire, nous avons dû faire des choix. Plusieurs idées intéressantes ont dû être mises de côté car le temps nous était compté. Nous pensons notamment à l'ajout d'un module de calcul pour les cordes, qui interviennent dans de nombreuses applications impliquant des membranes (tentes, parachutes, voiles, ...). Ces cordes pourraient, dans un premier temps, être considérées comme des membranes très fines avec une rigidité très élevée sous réserve de quelques hypothèses sur leur comportement, mais un modèle plus spécifique pourrait voir le jour.

Un modèle de froissement pourrait être introduit dans le code, afin d'affiner certaines modélisations. Nous pensons notamment aux coins d'un parachute, dans lesquels

des compressions transversales interviennent. Dans le modèle actuel, un raffinement local du maillage est nécessaire pour modéliser une zone comprimée.

Au niveau du couplage, il serait intéressant de mener une campagne d'expériences, non seulement pour valider le programme d'interaction, mais aussi parce que peu de travaux expérimentaux ont été réalisés à ce jour. Une autre vérification possible du couplage consisterait à comparer nos résultats avec un code dans lequel les forces seraient calculées par le programme structure et les déplacements par le programme fluide.

Enfin, certaines applications requièrent des conditions aux limites spécifiques au niveau du fluide, comme la modélisation d'écoulements sanguins rythmés par les pulsations cardiaques. Il serait possible d'introduire de telles conditions à l'avenir.

A ces pistes de réflexion s'ajoute la possibilité d'exploiter le programme produit pour étudier plus de cas pratiques ou examiner un problème particulier plus en profondeur.

Pour terminer nous ne résistons pas à l'envie d'exprimer notre espoir que d'autres étudiants utilisent et améliorent notre programme à l'occasion de travaux futurs, car sa réalisation fut une expérience unique, parsemée de moments d'étonnement, de joie, de déception, d'euphorie et de découvertes.

Bibliographie

- [1] *Jean-Frédéric Gerbeau, Marina Vidrascu, Pascal Frey*, Fluid-structure interaction in blood flows on geometries based on medical imaging, 05 mars 2004.
- [2] *Keith Stein, Tayfun Tezduyar, Richard Benney*, Applications in Airdrop Systems : Fluid-Structure Interaction Modeling, Fifth World Congress on Computational Mechanics, 7 juillet 2002.
- [3] *Kalro V., Tezduyar T.E.*, "A parallel 3D computational method for fluid-structure interactions in parachute systems", Computer methods in applied mechanics and engineering, 190 (2000) 321-332, 26 août 1999.
- [4] *Andrew J. Bax, Ronald A. Schachar*, Maximisation of the design of a scleral expansion band segment for the reversal of presbyopia, ANSYS Conference and Exhibition, 2000.
- [5] *Markus Glück, Michael Breuer, Franz Durst, Ansgar Halfmann, Ernst Rank*, Computation of Fluid-Structure Interaction on Lightweight Structures, Institute of Fluid Mechanics, University of Erlangen-Nürnberg.
- [6] *Chevaugéon N.*, Thèse, "Contribution à l'étude des membranes hyperélastiques en grandes déformations", Ecole centrale de Nantes et Université de Nantes, 14 janvier 2002.
- [7] *Mooney M.*, "A Theory of Large Elastic Deformation", Journal of Applied Physics, 11 (582-592), 1940.
- [8] *Rivlin R.*, "Large Elastic Deformations of Isotropic Materials I-IV", Philosophical Transactions of the Royal Society of London Series A, Mathematical and Physical Sciences, 240 (449-490), 1948.
- [9] *Hocine Bechir, Khaled Boufala, Yvon Chevalier*, Strain energy density function for carbon black filled rubber vulcanized for industrial applications, Mécanique et industries 3 (pg 245-252), 1 février 2002.
- [10] *Glenn K. Klute, Blake Hannaford, University of Washington*, Accounting for Elastic Energy Storage in McKibben Artificial Muscle Actuators.
- [11] *I. Ionescu, J. Guilkey, M. Berzins, R. M. Kirby, J. Weiss*, Computational Simulation of Penetrating Trauma in Biological Soft Tissues using the material Point Method.
- [12] *F. Erchiqui, A.N. Kandil*, Université de Québec Abitibi- témiscamingue, Neuronal Networks Application for Characterization of Softened Polymers.

- [13] *A. Masson, P. Sauvaget, O. Torlai, R. Thillou*, La modélisation hydraulique bi-dimensionnelle : un outil puissant d'analyse de scénarios d'aménagements et de restauration des cours d'eau.
- [14] *P.L. George* Transformation d'un maillage en un maillage aigu, Inria, France, le octobre 1992,
<http://www.edpsciences.org/articles/jp3/pdf/1993/01/jp3v3p55.pdf?access=ok>.
- [15] *Andréa Schwertner Charão* Multiprogrammation parallèle générique des méthodes de décomposition de domaine, 20 Septembre 2001.
- [16] *Pr Beckers*, Méthodes de discrétisation, ULg-LTAS,
<http://www.ulg.ac.be/ltras/publi/chapitre1.pdf>.
- [17] *V. Legat*, syllabus du cours d'*Introduction aux éléments finis*, UCL-FSA.
- [18] Dynamic analysis by numerical integration, Computers and structures INC, Oakland,
http://www.csiberkeley.com/Tech_Info/20.pdf.
- [19] *Jacques Rappaz*, Cours d'analyse numérique pour ingénieurs, Ecole Polytechnique Fédérale de Lausanne,
<http://dmawww.epfl.ch/rappaz.mosaic/Support/support/node72.html>.
- [20] *V. Legat et G. Winckelmans*, FSAC 1340 Méthodes numériques, UCL-FSA,
<http://www.mema.ucl.ac.be/vl/teaching/fsab1701/documents/cours6.pdf>.
- [21] *J Pellet* Notice de modélisation de l'amortissement mécanique, 08 novembre 02,
http://www.code-aster.org/DOCASTER/Man_U/U2/U20603b.pdf.
- [22] *C.H. Chen, Y.C. Wang*, An extended nonlinear mechanical model for solid-filled Mooney-Rivlin rubber composites, Polymer, vol 38 no. 3, 571–576, 1997.
- [23] Rubber elasticity, Mechanical Behaviour of Solids, University of Cambridge,
<http://www.msm.cam.ac.uk/Teaching/matmin1a/courseE/EP2.pdf>.
- [24] *Ingo Müller, Henning Struchtrup*, Inflating a Rubber Balloon, 31 may 2002.
- [25] *E. Verron, G. Marckmann*, Numerical analysis of rubber balloons, 22 janvier 2003.
- [26] *E. Verron, G. Marckmann*, An axisymmetric B-spline model for the non-linear inflation of rubberlike membranes, 17 juillet 2000.
- [27] *Geuzaine Philippe*, Site Internet du logiciel *Aero-F*,
<http://www.geuzaine.net/philippe/software/aerof3d.html>, 2003.
- [28] *Goffinet M., Marchandise E.*, Travail de fin d'études, "Développement d'un code CFD en volumes finis pour la modélisation 3D d'écoulements turbulents incompressibles.", UCL, 12 juin 2003.
- [29] *Fourestey Gilles, Pipernot Serge*, A second-order time-accurate ALE Lagrange-Galerkin method applied to wind engineering and control of bridge profile, Comput. Meths. Appl. Mech. Engrg., vol. 193, No. 39-41, pp. 4117-4137, 2004.
- [30] *Cadiou A.*, NadiaLES : Manuel théorique, Note technique CODICIEL-LMFA, N°2003-01,
http://www.codiciel.fr/plateforme/mecaflu/nadia/docs/manuel_theorique.pdf.
- [31] Centre hospitalier universitaire de Besançon, base de données multimedia,
<http://www.besancon-cardio.net/student/rep.php3>.

-
- [32] *Dr Gwenaël Kervadec*, Le système cardiovasculaire,
[http ://drgk.free.fr/biologie_humaine/04_text.htm](http://drgk.free.fr/biologie_humaine/04_text.htm).
- [33] Institut Mutualiste Montsouris, Pathologies cardiaques,
[http ://www.imm.fr/france/dep/ima/ima_int.htm](http://www.imm.fr/france/dep/ima/ima_int.htm).