THÈSE

préparée à

L'INRIA Sophia Antipolis

et présentée à

L'UNIVERSITÉ de NICE - SOPHIA ANTIPOLIS

pour obtenir le grade de

DOCTEUR

Spécialité

Sciences de l'Ingénieur

par

Guillaume PICINBONO

Sujet de la thèse :

Modèles géométriques et physiques pour la simulation d'interventions chirurgicales

Soutenance prévue le 12 février 2001 devant un jury composé de :

Pierre	**Bernhard**	Président
Philippe	**Cinquin**	Rapporteurs
Daniel	**Thalmann**	
Nicholas	**Ayache**	Directeur
Hervé	**Delingette**	Co-Directeur
Jacques	**Marescaux**	Examinateurs
Marina	**Vidrascu**	

À Caro,
et à ma famille,

Remerciements

Je tiens tout d'abord à remercier chaleureusement Nicholas Ayache, mon directeur de thèse, de m'avoir accueilli au sein du projet Epidaure et m'avoir aiguillé sur un sujet de recherche passionnant. Je remercie également Hervé Delingette, qui a encadré de près tous mes travaux, pour sa disponibilité et sa rigueur scientifique. Un grand merci à Jean-Christophe Lombardo, avec qui j'ai eu grand plaisir à collaborer durant les deux années de l'AISIM, et qui m'a fait profiter de sa grande expérience en informatique.

Je voudrais aussi remercier les membres du jury : Philippe Cinquin et Daniel Thalmann qui ont accepté la lourde tache d'être les rapporteurs de mes travaux; Marina Vidrascu, pour ses conseils, ses explications et ses encouragements; Jacques Marescaux, dont l'enthousiasme est une source de motivation intarissable; et enfin Pierre Bernhard, président du jury, dont l'esprit et la bonne humeur ont su dédramatiser la difficile épreuve de la soutenance.

Merci à tous les membres de l'équipe Epidaure : Grégoire Malandin, Éric Bardinet, Xavier Pennec et Françoise Pezé pour leur aide et leur soutien; l'irremplaçable Janet Bertot, ingénieur système de son état, pour son aide précieuse, sa disponibilité inconditionnelle et les nombreux cafés que nous avons partagés. Un grand merci à Johan Montagnat, pour sa pédagogie et son incomparable "yav++". Merci à mes compagnons de thèse de la cuvée 2001, Alexis et Sylvain, avec qui j'ai partagé les difficultés de la rédaction et les longs week-end à l'INRIA. Merci à tous les doctorants : David, collègue de labeur du bureau F214 durant deux années, mais aussi de détente lors d'inoubliables parties de ping-pong en compagnie de Pascal, Jonathan, Yves et Oliver; merci à Clément pour son aide à l'approche de ma soutenance, merci à Sébastien de m'avoir ramené mon échelle, et à tous les autres pour la bonne ambiance générale au sein du labo et plus particulièrement autour de la machine à café.

Je voudrais également exprimer toute ma gratitude à l'équipe du service informatique de l'INRIA, à Francis Montagnac dont j'ai pu à maintes reprises apprécier les compétences, et aux "filles de doc" pour leur aide et leur efficacité.

Enfin, je tiens à remercier ma famille et mes amis pour leur soutien de tous les instants. Merci papa pour ces parties de pêche aux pageots, maman et Cécile pour votre relecture du manuscrit et Mathilde pour ton soutien psychologique à distance. Merci à tous pour votre présence réconfortante à la soutenance et pour la préparation du pot de thèse.
Merci Caro pour tout ce bonheur ...

Table des matières

Introduction

Le thème des travaux que nous allons présenter dans ce manuscrit est la simulation de chirurgie laparoscopique. Après avoir mis en évidence les avantages et les difficultés liés à ce type de chirurgie, nous présenterons les intérêts, ainsi que le schéma général de fonctionnement, d'un simulateur de chirurgie en réalité virtuelle. Nous poursuivrons par un état de l'art du domaine, avant de donner un aperçu des travaux effectués durant cette thèse en insistant sur les contributions originales que nous proposons.

1 Le cadre général

Durant ces vingt dernières années, les technologies liées à la médecine ont énormément progressé. L'avènement de nouveaux types d'imagerie et de nouvelles techniques d'acquisition permet de fournir aux médecins toujours plus d'informations qui les aideront dans leurs diagnostics. Les progrès réalisés dans les domaines de la vidéo (qualité et miniaturisation des caméras, utilisation des fibres optiques), des télécommunications et de la robotique donnent le jour à de nouvelles méthodes de diagnostic (coelioscopie, échographie exploratrice) et à de nouvelles techniques de chirurgie (vidéo-chirurgie, télé-chirurgie, chirurgie assistée par ordinateur) (Cinquin et Troccaz, 1999). Parallèlement, les progrès phénoménaux réalisés sur les ordinateurs, dont les puissances de calcul et les capacités de mémoire ne cessent d'augmenter, permettent aujourd'hui d'apporter une aide conséquente au monde médical. Le champ d'action de l'informatique médicale est relativement large. Un premier grand thème est celui de la visualisation et du traitement des données tridimensionnelles acquises par différentes modalités d'imagerie (scanner, IRM [1], imagerie nucléaire, échographie) (Ayache, 1998; Duncan et Ayache, 2000). Le médecin peut alors visualiser de manière beaucoup plus naturelle les images 3D ou même 4D (séquence d'images acquises au cours du temps). Les outils informatiques permettent également d'appliquer des traitements sur ces images, afin d'en améliorer la qualité (suppression du bruit, augmentation du contraste, rehaussement des contours) (Krissian, 2000). Il est aussi possible de les segmenter, c'est-à-dire de délimiter des zones d'intérêt comme les organes (Montagnat, 1999; Delingette, 1994), les vaisseaux sanguins ou les tumeurs (Soler, 1998). On fournit ainsi au médecin une information anatomique ou pathologique supplémentaire qu'il pourra utiliser lors de son interprétation des données. Dans le cas de certaines maladies évolutives (cancer, sclérose en plaque), le traitement d'une séquence d'images permet d'identifier les lésions et d'en quantifier l'évolution (Rey et al., 1999). Ce résultat peut ensuite être corrélé avec les symptômes du patient ou le traitement qu'il suit. Une autre application intéressante est le recalage d'images (Cachier et Pennec, 2000; Cachier et Rey, 2000; Roche et al., 2000), qui consiste à replacer un ensemble d'images dans le même référentiel afin d'en faciliter la comparaison. L'informatique peut aussi apporter une aide précieuse aux chirurgiens. Par exemple en neurochirurgie, des logiciels permettent de déterminer très précisément la trajectoire selon laquelle doit être introduit un instrument dans le cerveau afin d'éviter les zones vitales. Une fois la trajectoire définie le programme pourra guider un robot ou bien aider le chirurgien à effectuer le geste (réalité augmentée).

Enfin, un autre domaine de prédilection de l'informatique appliquée à la médecine est celui de la simulation. Il s'agit dans ce cas de fournir aux chirurgiens ou aux médecins (en particulier les radiologues, qui sont amenés à réaliser des procédures relativement complexes), un outil leur permettant de s'entraîner, de répéter des gestes difficiles ou d'essayer et de valider de nouveaux enchaînements. C'est dans ce dernier cadre que s'inscrivent les travaux de cette thèse : développer un ensemble d'outils informatiques, basés sur les principes de la physique, des mathématiques et de la géométrie, qui constitueront les pièces nécessaires à l'élaboration d'un simulateur de chirurgie.

1. Imagerie par Résonance Magnétique

Les travaux sur la simulation de chirurgie ont débuté au sein du projet Epidaure en 1994, avec la thèse de Stéphane Cotin (Cotin, 1997). Ces recherches s'inscrivaient alors dans le cadre du projet européen MASTER[2], dans lequel coopéraient des instituts de recherches, des industriels et des établissements hospitaliers. Les travaux ont débouché sur l'élaboration d'un prototype de simulateur de chirurgie laparoscopique sur le foie, qui a servi de point de départ aux développements présentés dans ce document.

Ce thème de recherche a pris de l'ampleur à l'INRIA en 1997, avec la mise en place d'une "Action de Recherche Collaborative". Ainsi, l'AISIM[3] a permis la collaboration, durant deux années, de 6 projets de l'INRIA (les projets Epidaure et Sinus à Sophia Antipolis, iMAGIS et SHARP à Grenoble et M3N et MACS à Rocquencourt) et d'un expert médical représenté par l'IRCAD[4], localisé à Strasbourg. Cette pluridisciplinarité a permis de s'attaquer à de nombreux problèmes clés liés à la simulation de chirurgie, souvent avec des approches concurrentes ou complémentaires. Les principaux sujets traités furent la modélisation réaliste du comportement biomécanique du foie (sans contrainte de temps-réel), l'élaboration de modèles déformables temps-réels du foie, le rendu réaliste, et les problèmes liés à l'utilisation de systèmes à retour d'effort. Nous reviendrons plus en détails sur les travaux réalisés durant ces deux années dans la partie consacrée à l'état de l'art.

En 1999 a débuté CAESARE[5], qui est une action concertée incitative financée par le Ministère de l'éducation nationale, de la recherche et de la technologie. Ce projet a été sélectionné dans le cadre de l'appel d'offres concernant la "Télémédecine et Technologie pour la Santé". CAESARE est la prolongation directe de l'AISIM, et sa durée est aussi de deux ans. Il regroupe d'ailleurs les mêmes partenaires, auxquels vient s'ajouter la société ESI[6]. Les thèmes de recherche sont les mêmes, mais avec en plus la finalité de développer un prototype de simulateur en collaboration avec ESI, en vue d'une évaluation au sein de l'EITS[7], un centre de formation des chirurgiens rattaché à l'IRCAD.

Les travaux présentés dans cette thèse ont donc été réalisés durant les deux années qu'a duré l'AISIM et se sont prolongés pendant la première année de CAESARE. Le thème central de ces recherches étant la chirurgie dite "minimalement invasive", nous allons maintenant en décrire les principes, les avantages et les inconvénients.

2 La chirurgie minimalement invasive

Le terme de thérapie "minimalement invasive" regroupe toutes les techniques de diagnostic ou de soin visant à minimiser les traumatismes occasionnés à l'organisme du patient. Dans le cadre de la chirurgie, le point clé est de limiter le nombre et surtout la taille des incisions. Nous nous intéressons plus particulièrement à la chirurgie abdominale. La chirurgie abdominale en laparoscopie consiste à

2. Minimal Access Surgery by Telecommunications and Robotics
3. Action Incitative SIMulation de chirurgie : http://www.inria.fr/epidaure/AISIM
4. Institut de Recherche sur le Cancer de l'Appareil Digestif : http://www.ircad.org/
5. Chirurgie Abdominale Et Simulation A Retour d'Effort : http://www-sop.inria.fr/epidaure/CAESARE/
6. ESI : http://www.esi.fr/
7. European Institute of TeleSurgery

introduire des trocarts (voir figure 1(b)) au travers de petits trous (environ 1 centimètre) pratiqués dans l'abdomen du patient, au lieu d'effectuer une large incision qui peut faire plusieurs dizaines de centimètres. Par l'intermédiaire de ces trocarts, on introduit à l'intérieur de l'abdomen du patient l'optique d'une caméra (qui est aussi une source de lumière), et des instruments spéciaux (figure 1(a)).

(a) Instruments utilisés en chirurgie laparoscopique : plusieurs formes de pinces et de ciseaux sont disponibles.

(b) Trocarts de 5 et 10 mm de diamètre. Le plus gros est utilisé pour introduire la caméra ou certains gros instruments. L'autre est réservé aux outils les plus simples.

FIG. 1 – *Trocarts et instruments utilisés pour la chirurgie abdominale en laparoscopie.*

L'intérieur de l'abdomen du patient, filmé par la caméra, est retransmis sur un écran haute définition. Cette caméra est manipulée par un assistant. L'abdomen du patient est gonflé de gaz neutre (CO_2), afin de permettre une meilleure visibilité. Le chirurgien va donc opérer à l'aide d'instruments qui passent à travers l'abdomen du patient, en visionnant la scène sur un écran vidéo (figure 2).

Les principaux avantages sont évidemment pour le patient. Les traumatismes occasionnés à l'organisme étant moins importants, le patient se remet plus vite de l'opération, souffre moins et ne garde que de petites cicatrices. Le risque de complications est aussi nettement diminué puisqu'à aucun moment l'intérieur du corps humain n'est en contact direct avec l'air. En conséquence, le temps d'hospitalisation est diminué, ce qui diminue le coût total des soins médicaux.

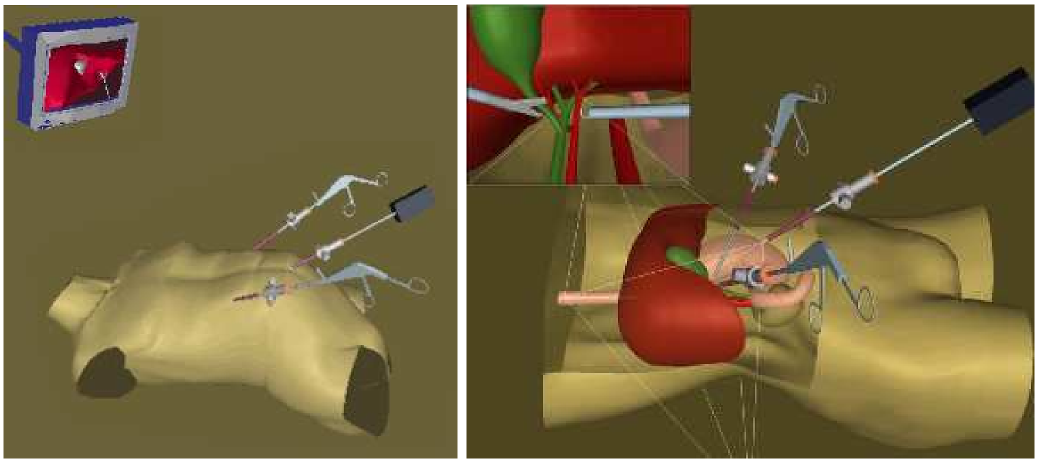


FIG. 2 – *Principes de la chirurgie abdominale en laparoscopie (© KISMET Medical Applications).*

Par contre, les principales difficultés se posent au chirurgien. Il doit d'abord apprendre les gestes liés à ce nouveau type de chirurgie et doit aussi s'adapter au nouvel environnement dans lequel il va opérer. La figure 3 montre les difficultés que doit affronter le chirurgien :

- Tout d'abord, le chirurgien a troqué ses dix doigts contre deux longs instruments dont les déplacements sont contraints par le passage par un point fixe (le trocart).
- Ensuite, il visualise l'opération sur un écran vidéo, ce qui entraîne deux difficultés : il perd la troisième dimension, ce qui pose des problèmes de perception de la profondeur, et il ne regarde plus à l'endroit où se passe l'opération puisque le moniteur est placé en hauteur afin que toutes les personnes participant à l'opération puissent suivre son déroulement. Ce dernier point oblige le chirurgien à désynchroniser sa vue de ses gestes.
- Enfin, en général, la caméra n'est pas manipulée par le chirurgien mais par un assistant ou par un bras robotisé à commande vocale. Cela oblige le chirurgien à demander que le point de vue soit modifié, alors qu'il lui suffirait, en chirurgie classique, de bouger la tête.

FIG. 3 – *Chirurgie abdominale en laparoscopie (© 1997 United States Surgical Corporation).*

Toutes ces difficultés obligent le chirurgien à effectuer de nombreuses heures d'entraînement, afin d'acquérir la dextérité supplémentaire requise pour la manipulation des instruments, pour réussir à reconstruire mentalement la scène en trois dimensions à partir des indices disponibles (ombres, reflets, textures des organes), et pour apprendre une nouvelle coordination de l'œil et de la main.

Aujourd'hui, cet entraînement se fait principalement de deux façons :

- La première possibilité est d'utiliser un simulateur mécanique, ou *endotrainer*. Ce type simulateur de est intéressant pour apprendre à manipuler les instruments chirurgicaux et à effectuer les gestes de base comme les nœuds et les sutures. Par contre, il propose en général une représentation peu réaliste du comportement biomécanique et physiologique des organes.

FIG. 4 – *Simulateur mécanique de chirurgie laparoscopique (© 1997 United States Surgical Corporation).*

- La seconde possibilité est de s'entraîner sur des animaux. Pour la chirurgie abdominale, cet entraînement se fait sur des cochons, car la forme et la répartition des organes abdominaux sont proches de ceux de l'homme. L'EITS propose des cours de formation à la chirurgie laparoscopique, dont la partie pratique se fait de cette façon. Mais ce genre formation coûte très cher et pose quelques problèmes éthiques.

FIG. 5 – *Entraînement sur les cochons ...*

Au final, ces deux méthodes d'entraînement ne représentent que peu d'heures, et le véritable apprentissage se fait sur des patients, sous le contrôle de chirurgiens expérimentés. Il existe donc un réel besoin de compléter la formation des chirurgiens à ces nouvelles techniques, que seul un simulateur de chirurgie laparoscopique réaliste peut combler. Si aujourd'hui tous les pilotes d'avions ont reçu une formation sur simulateur, on imagine facilement qu'il en sera de même pour les chirurgiens dans les années à venir. Il existe d'ailleurs déjà des produits commercialisés que nous décrirons dans le paragraphe consacré l'état de l'art (5.2).

3 Intérêt d'un simulateur en réalité virtuelle

Un simulateur de chirurgie en réalité virtuelle peut être utilisé dans plusieurs contextes différents. Le premier est, bien évidement, la formation et l'entraînement aux gestes. Dans ce cadre, on parlera de simulateurs généraux, dans la mesure où ils sont sensés représenter un patient moyen avec une anatomie "d'école". Ils permettront aux chirurgiens de se familiariser avec un environnement nouveau et d'acquérir les réflexes correspondants.

Mais on peut aussi envisager des simulateurs spécifiques, qui intégreront l'anatomie et les caractéristiques d'un patient particulier. Par exemple dans le cas de la chirurgie hépatique, il est envisageable de reconstruire à partir du scanner du patient la géométrie du foie et des organes environnants (Montagnat et Delingette, 1998; Montagnat, 1999). On pourra ensuite reconstruire le réseau vasculaire hépatique, en déduire la subdivision du foie en segments, et détecter les éventuelles lésions (tumeurs, kystes) (Soler, 1998; Soler et al., 2000b; Soler et al., 2000a). Toutes ces informations pourront alors être regroupées dans un modèle déformable respectant les particularités du patient. Le chirurgien pourra alors, dans d'excellentes conditions, préparer son intervention et décider d'une stratégie opératoire. On parlera alors de planification chirurgicale.

Dans les deux cas, un des grands avantages d'un simulateur en réalité virtuelle sera de pouvoir provoquer des évènements inattendus à n'importe quel instant. L'élève chirurgien pourra donc être confronté à des difficultés supplémentaires qui ne sont rencontrées que très rarement lors des opérations réelles, mais auxquelles il doit être préparé. Ces situations particulières peuvent être provoquées par un second utilisateur, ou bien simplement de manière aléatoire.

Enfin, un simulateur de chirurgie peut aussi permettre d'évaluer les qualités et les défauts d'un geste à partir de critères précis. On pourra par exemple surveiller la vitesse de déplacements des instruments, la précision de certains gestes ou l'intensité des contraintes appliquées sur l'organe virtuel.

4 Schéma général de fonctionnement

Un certain nombre de simulateurs de chirurgie ont vu le jour ces dernières années. Si les buts recherchés et les applications peuvent être très différents, l'architecture et le schéma général de fonctionnement de ces simulateurs sont toujours à peu près les mêmes. La pièce maîtresse est le modèle déformable représentant la géométrie et le comportement biomécanique de l'organe à opérer. La seconde partie du simulateur est constituée d'un moyen d'interaction entre l'utilisateur et le modèle déformable. Cette interaction se fera en général par l'intermédiaire d'une interface mécanique, équipée ou non d'un dispositif à retour d'effort. Dans le cas de notre simulateur de chirurgie hépatique en laparoscopie, les deux parties principales seront un modèle déformable représentant un foie et une interface à retour d'effort respectant l'ergonomie des instruments utilisés par les chirurgiens. D'un point de vue logiciel, ces deux composantes sont réparties dans deux boucles asynchrones (figure 6). La boucle gérant l'interface à retour d'effort envoie la position et récupère les forces qu'elle doit appliquer. Pour obtenir un retour d'effort réaliste, cette boucle doit fonctionner à une fréquence de l'ordre de 300 Hz. La seconde boucle récupère la position de l'instrument, détecte les collisions entre l'instrument et le modèle déformable, traduit ces collisions en contraintes qui sont appliquées au modèle déformable afin de modéliser le contact, calcule la déformation et la force exercée sur l'instrument. Cette force est renvoyée au système à retour d'effort. Enfin, cette boucle met à jour l'affichage de la scène à une fréquence de 25 Hz.

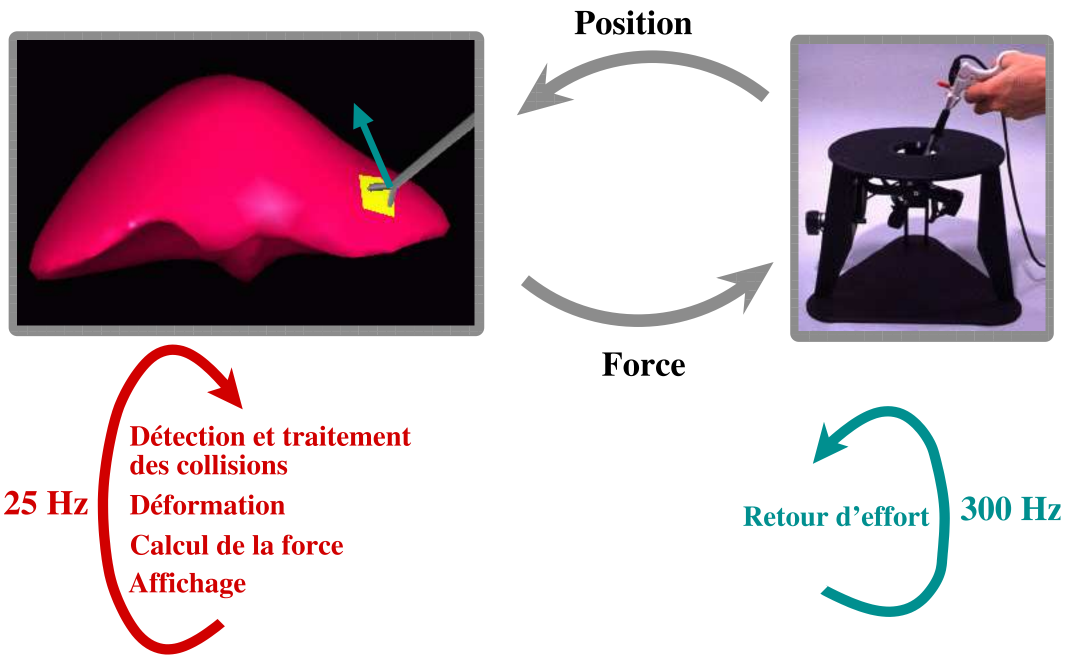


FIG. 6 – *Architecture du simulateur de chirurgie laparoscopique sur le foie.*

4.1 Le modèle déformable

Le point clé d'un simulateur de chirurgie est l'organe virtuel. Ce modèle déformable doit être capable de représenter, avec le plus de réalisme possible, à la fois la forme et le comportement biomécanique de l'organe que l'on désire représenter. Il doit, en outre, pouvoir subir des interactions avec les organes environnants ou les instruments de chirurgie. La plupart de nos travaux sont réalisés sur le foie. La forme de l'organe est extraite à partir du scanner de patient à l'aide de surfaces déformables qui vont, sous l'action de contraintes adéquates, venir délimiter les contours du foie ((Montagnat et Delingette, 1998)). À partir de ce maillage de la surface, on construit un maillage volumique. La troisième étape consiste à associer une loi de déformation à ce maillage, c'est-à-dire définir de quelle manière l'organe doit se déformer lorsqu'une contrainte lui est appliquée. Les paramètres de la loi de déformation choisie doivent être identifiés aux caractéristiques biomécaniques mesurés sur l'organe lui-même (voir partie sur l'anatomie et la rhéologie du foie, chapitre 5). Enfin, on doit définir quelles sont les contraintes externes que l'on peut appliquer à notre modèle. En général, on veut être capable d'appliquer des déplacements ou des forces sur une partie ou sur la totalité de l'organe. Quand le modèle le permet, on veut aussi pouvoir modifier sa topologie (raffiner le modèle localement, ou bien le découper).

4.2 Interaction avec le modèle déformable

L'interaction avec le modèle déformable comprend deux parties importantes. D'abord il faut coupler au simulateur une interface mécanique qui permettra à l'utilisateur d'interagir avec l'environnement virtuel. Cette interface peut être équipée d'un dispositif à retour d'effort qui augmente sensiblement le réalisme du simulateur. La seconde partie est plus spécifique au but du simulateur. Il s'agit de modéliser la façon dont l'instrument virtuel manipulé par l'utilisateur va interagir avec le modèle déformable. Il faudra pour cela définir les notions de contact et de glissement, et prévoir un certain nombre de gestes de base (agripper, glisser, couper, ...).

5 Simulation de chirurgie : état de l'art

Comme nous venons de le souligner, le modèle déformable est le centre névralgique d'un simulateur de chirurgie. Son choix est dicté par le cahier des charges du simulateur, et en particulier par le compromis entre le réalisme et l'interactivité. En retour, une fois choisi, c'est le modèle déformable qui conditionnera les possibilités, et les améliorations possibles, du simulateur tout entier. Nous allons donc commencer cet état de l'art par un tour d'horizon des différents modèles déformables utilisés pour représenter le comportement d'objets mous et de structures anatomiques.

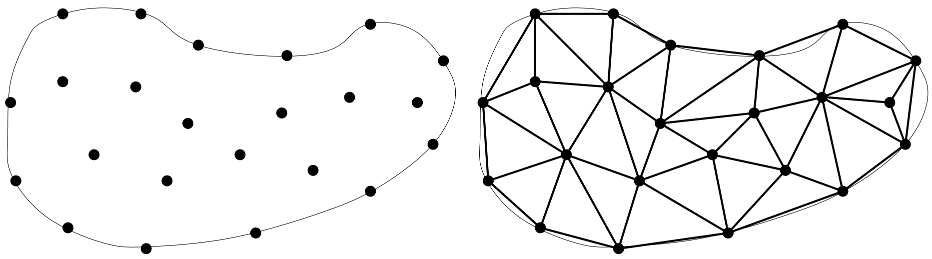
5.1 Les modèles déformables physiques

Dans un simulateur de chirurgie, le modèle déformable devra représenter un organe ou une structure anatomique, c'est-à-dire un certain volume de tissu biologique mou que l'on voudra toucher, déformer et couper. Il doit donc garantir un certain réalisme biomécanique. C'est pourquoi la plupart des simulateurs mettent en œuvre des modèles déformables "physiques", c'est-à-dire dont les déformations sont dérivées des lois de la mécanique des milieux continus. Si de tels modèles sont utilisés depuis longtemps pour le calcul de structures et pour la modélisation des interactions entre fluides et solides, leur utilisation, pour l'animation ou pour des applications temps-réel comme la simulation, est plus récente, en particulier sous l'impulsion des travaux de D. Terzopoulos[8] (Terzopoulos et al., 1987; Terzopoulos et Fleicher, 1988).

Les modèles déformables qui nous intéressent résultent de l'approximation géométrique et physique de l'objet que l'on veut représenter. Le degré et la nature des approximations réalisés conditionnent le réalisme du modèle obtenu. Il est en général impossible de résoudre les équations de la mécanique de manière analytique. On doit alors avoir recours à l'analyse numérique pour trouver

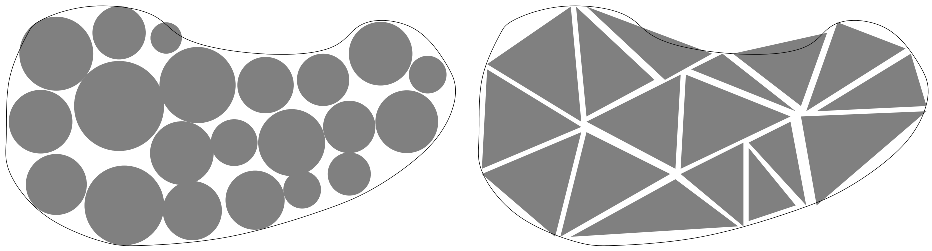
8. D. Terzopoulos : http://www.cs.toronto.edu/~dt/main.html

une solution approchée au problème posé. Or tout calcul numérique repose sur une discrétisation du domaine sur lequel on travaille. C'est à ce moment qu'intervient l'approximation géométrique du domaine. Il existe plusieurs façons de représenter de manière discrète un domaine. La première étape est de le diviser en sous-domaines élémentaires. Par exemple en dimension 2, un objet peut être représenté par un ensemble de points, de sphères, ou de polygones (figure 7). Cette étape per-



(a) Approximation par un ensemble de points.

(b) Approximation par un ensemble de points et de segments.



(c) Approximation par un ensemble de sphères.

(d) Approximation par un ensemble de triangles.

FIG. 7 – *Différentes approximations géométriques d'un domaine.*

met donc de définir les formes de base sur lesquelles les équations du problème seront approximées, ce qui correspond à fixer le nombre de degrés de liberté autorisés. La qualité de la discrétisation dépend, bien entendu, du nombre de sous-domaines que l'on s'autorise pour décrire l'objet. On peut ensuite différencier les modèles géométriques structurés, c'est-à-dire ceux où les sous-domaines ont des liens topologiques avec leurs voisins, des modèles non structurés, où les éléments peuvent se déplacer librement les uns par rapport aux autres.

5.1.1 Le modèle Masse-ressort

Lorsque l'on s'attache à représenter un objet mou, le modèle le plus simple, mais aussi le plus intuitif, est sans doute le réseau masse-ressort. Il consiste à modéliser l'objet par un ensemble de points reliés entre eux par des segments (figure 7(b)). Les points représentent des masses considérées comme ponctuelles, et les segments sont considérés comme des ressorts. À chaque itération, on calcule les forces exercées par chaque ressort sur les deux masses situées à ses extrémités. Ces forces s'expriment en fonction de la variation de longueur du ressort et de sa raideur. On peut ensuite calculer les nouvelles positions des masses en intégrant leurs équations du mouvement. Il y a plusieurs raisons au succès de ce modèle. Il est tout d'abord très efficace car les calculs sont très simples. Cela permet notamment d'utiliser plusieurs modèles masse-ressort pour simuler une scène complexe. Par exemple, L. Nedel [9] modélise les muscles d'un humain virtuel avec des réseaux masse-ressort surfaciques (Nedel et Thalmann, 1998b; Nedel et Thalmann, 1998a). En plus, le coté intuitif de ce modèle permet de lui apporter de nombreuses améliorations. Par exemple, on peut jouer sur le comportement des ressorts et sur leurs raideurs pour construire des modèles non-linéaires et non-homogènes (d'Aulignac et al., 1999a; Boux de Casson et Laugier, 1999), ou anisotropes (Bourguignon et Cani, 2000). Par contre, le principal défaut de ce modèle est de discrétiser les directions selon lesquelles les forces s'exercent à l'intérieur du matériau. En effet, les forces sont forcement dirigées suivant les arêtes qui composent le maillage géométrique. L'implication la plus pénalisante de cette propriété est que le comportement global de l'objet simulé va dépendre de la géométrie du maillage, et plus particulièrement de la répartition des directions des arêtes. De nombreux travaux ont d'ailleurs portés sur l'amélioration du comportement de ce modèle. Par exemple, Y. Chen propose un réseau masse-ressort relativement sophistiqué comprenant différents types de ressorts (ressorts structurels, ressorts de cisaillement et ressorts de flexion) afin de simuler la déformation que subit un muscle lors de sa contraction (Chen et al., 1998). Il est aussi possible d'améliorer le réalisme d'un modèle masse-ressort en donnant aux ressorts un comportement déduit de mesures expérimentales (Boux de Casson, 2000), ou en adaptant les raideurs des ressorts afin que le modèle atteigne le comportement désiré (d'Aulignac et al., 1999b). O. Deussen [10] propose des méthodes pour améliorer un modèle masse-ressort dans plusieurs directions (Deussen et al., 1995). Il commence par optimiser la répartition des sommets du maillage à l'aide d'une décomposition itérative du domaine en diagrammes de Voronoy. Il calcule ensuite la valeur des masses attribuées à chaque sommet de manière à conserver les moments d'inertie d'ordres 1 et 2 à l'intérieur du modèle. Enfin, il déduit les raideurs des ressorts en identifiant son modèle avec un modèle élastique linéaire. Dans le même ordre d'idée, A. Gelder [11] propose une formule permettant de calculer la raideur des ressorts de manière à ce que le modèle masse-ressort se rapproche le plus possible de son homologue élastique linéaire discrétisé par la méthode des éléments finis (Gelder et Wilhelms, 1997). C'est d'ailleurs ce que nous utiliserons paragraphe 2.5.2. Enfin, on peut citer les travaux de M. Desbrun [12], qui propose un algorithme d'intégration permettant de stabiliser et d'accélérer le calcul des déformations d'un modèle masse-ressort(Desbrun et al., 1999).

9. L. Nedel : http://www.inf.ufrgs.br/~nedel/
10. O. Deussen : http://www.inf.tu-dresden.de/~od1/
11. A. Gelder : http://www.cse.ucsc.edu/~avg/
12. M. Desbrun : http://www.multires.caltech.edu/~mathieu/

5.1.2 Les systèmes de particules

Le principe des modèles déformables à base de particules est de mimer le comportement de la matière à l'échelle atomique. Les particules sont des masses ponctuelles qui interagissent entre elles par l'intermédiaire d'un champ d'attraction-répulsion (répulsion à courte portée et attraction à longue portée). Les objets modélisés par un système de particules sont souvent représentés par des surfaces implicites (surfaces équipotentielles d'une fonction, qui, dans ce cas, sera la fonction représentant le champ d'interaction entre les particules). Un système de particules n'utilise pas forcement un modèle géométrique structuré. Les particules ont des interactions avec leur voisines, mais ne sont pas contraintes à conserver un certain voisinage. A. Murta[13] met à profit cette propriété pour modéliser des liquides en mouvements (Murta et Miller, 1999). Mais il est aussi possible d'utiliser un système de particules pour représenter des corps déformables. F. Jaillet[14] propose, dans sa thèse, un état de l'art particulièrement complet sur ce sujet (Jaillet, 1999). Ses travaux portent sur la reconstruction d'objets volumiques à partir de données structurées en coupes, la principale application étant l'imagerie médicale. Il modélise ensuite les déformations des objets reconstruits à l'aide de systèmes de particules (Jaillet et al., 1997).

Le principal défaut des systèmes de particules est leur temps de calcul qui varie en $O(n^2)$ (n étant le nombre de particules), puisqu'il faut calculer, à chaque instant, les forces exercées par chaque particule sur toutes les autres. M.-P. Cani[15] et M. Desbrun proposent une solution à ce problème en imposant un rayon d'action aux particules afin de limiter le nombre d'interactions. Ce modèle leur permet de représenter des matériaux fortement déformables (Desbrun et Gascuel, 1995). Ils définissent de plus un nouveau type de particules inspirées des *Smoothed Particules Hydrodynamics* (Monaghan, 1992) utilisées en astrophysique, qui permet d'animer aussi bien des fluides que des solides (Desbrun et Gascuel, 1996; Desbrun, 1997).

Si l'on veut simplifier encore un peu plus un système de particules, on peut restreindre l'action d'une particule à un certain nombre de ses voisines les plus proches, que l'on désigne une fois pour toutes pour ne plus avoir à les retrouver. Cela revient alors à définir une notion de voisinage pour les particules et à exprimer la force qui les lie en fonction de la distance qui les sépare. On se ramène alors à un modèle masse-ressort.

5.1.3 Les modèles élastiques

La théorie de l'élasticité permet de décrire le comportement de tout matériau déformable, qu'il soit homogène ou pas, linéaire ou non-linéaire, isotrope, anisotrope, ou encore plus ou moins incompressible (Ciarlet, 1988; Ogden, 1984). Elle repose sur la relation existant entre la déformation subie par chaque élément de volume et les contraintes qu'il subit. L'élasticité constitue donc naturellement la voie reine pour décrire les déformations de matériaux biologiques dont le comportement

13. A. Murta : http://www.cs.man.ac.uk/~amurta/alan-murta.html
14. F. Jaillet : http://www710.univ-lyon1.fr/~fjaillet/
15. M.-P. Cani : http://www-imagis.imag.fr/~Marie-Paule.Cani-Gascuel/mpg_fr.html

est, la plupart du temps, complexe. Malheureusement, les équations mises en jeu sont souvent d'une grande complexité et leur résolution est particulièrement coûteuse en temps de calcul. La méthode des éléments finis est souvent utilisée pour résoudre les problèmes d'élasticité (Zienkiewicz, 1977). Son principal avantage est de permettre une description continue à la fois du problème et de sa solution. Mais il est aussi possible d'utiliser les méthodes des différences finies ou des volumes finis (Debunne et al., 1999). Nous décrirons de manière plus précise la théorie de l'élasticité et la méthode des éléments finis dans le chapitre 1.

Les modèles élastiques permettent de décrire avec beaucoup de réalisme le comportement biomécanique de nombreux organes et tissus biologiques. Ainsi, le *Visualization Lab*[16] de l'université de New York travaille sur la modélisation des muscles, qui sont reconstruits à partir des données du *Visible Man*. Le muscle est ensuite animé grâce à un modèle physiologique de contraction. La déformation du muscle est obtenue grâce à un modèle élastique linéaire utilisant la méthode des éléments finis (Zhu et al., 1998). Une autre application intéressante est proposée par F. Azar[17] : il a développé un modèle déformable du sein qui lui permet de localiser des tumeurs sur différents examens au cours desquels le sein se déforme en fonction de la position de la patiente et des conditions d'acquisition des images. Il utilise pour cela un maillage hexaédrique relié à une surface triangulée. Le comportement physique est approximé par une loi d'élasticité linéaire par morceaux : la pente de la loi linéaire est réévaluée à chaque itération à partir d'une courbe contrainte-déformation non-linéaire expérimentale (Azar et al., 2000).

On trouve aussi de nombreux travaux sur l'étude des déformations du cerveau qui mettent en œuvre des modèles élastiques et la méthode des éléments finis (Martin et al., 1998). M. Ferrant[18] utilise un modèle élastique linéaire du cerveau, pour le recalage non-rigide d'IRM (Ferrant et al., 1999) et pour la modélisation du *brain shift*[19] (Ferrant et al., 2000b; Ferrant et al., 2000a). M. Miga, pour sa part, propose un modèle plus sophistiqué du cerveau basé sur la physique de la consolidation. On peut le voir comme un modèle élastique linéaire non-homogène et incompressible, qui tient compte des interactions entre fluide (liquide céphalo-rachidien) et structure (matières, blanche et grise, du cerveau). Il l'utilise pour modéliser le *brain shift*, ainsi que les déformations dues à l'apparition ou à la suppression de tumeurs (Miga et al., 1997; Miga et al., 1998b; Miga et al., 1998a; Miga et al., 2000; Paulsen et al., 1999). Enfin, C. Davatzicos[20] a développé un modèle élastique linéaire non-homogène précontraint avec lequel il opère le recalage non-rigide d'IRM du cerveau (Davatzikos, 1996; Davatzikos, 1997).

---

16. Visualization Lab : http://www.cs.sunysb.edu/~vislab/vislab_home.html
17. F. Azar : http://surf.to/fredazar
18. M. Ferrant : http://www.tele.ucl.ac.be/MEMBERS/Ferrant_Matthieu_e.html
19. Déformation subie par le cerveau lorsque la boite crânienne est ouverte et qu'une partie du liquide céphalo-rachidien dans lequel flotte le cerveau s'est échappée.
20. C. Davatzicos : http://ditzel.rad.jhu.edu/~hristos/html/christos_bio.html

5.2 Les simulateurs

Au fur et à mesure que la puissance des ordinateurs augmente, de nouveaux simulateurs de chirurgie voient le jour, utilisant des modèles déformables toujours plus sophistiqués couplés à des effets spéciaux qui améliorent grandement le réalisme. Nous proposons ici un tour d'horizon de l'état de l'art aujourd'hui.

Il existe déjà un certain nombre de sociétés qui ont mis sur le marché des simulateurs de chirurgie. C'est le cas de **ReachIn**[21], qui commercialise un simulateur de chirurgie laparoscopique avec retour d'effort, qui peut être utilisé pour un grand nombre d'applications. Ils proposent aussi des librairies de programmes et même leurs interfaces à retour d'effort. La société **Xitact**[22] a développé un produit, nommé *Virtual Patient*, permettant de simuler des gestes de chirurgie minimalement invasive avec retour d'effort. Ce simulateur est issu des recherches menées à l'Institut de Systèmes Robotiques[23] de Lausanne (EPFL). La société **Simbionix**[24] commercialise, elle aussi, un simulateur de chirurgie endoscopique. Des applications à l'urologie endoscopique, la gynécologie, la radiologie interventionnelle et la laparoscopie sont également en cours de développement. On peut aussi citer la société **Boston Dynamics**[25], dont les simulateurs de chirurgie avec retour d'effort permettent la simulation d'arthroscopie[26] et d'anastomose[27] en chirurgie ouverte (suture de structure tubulaire). Dans chacun des cas la simulation est accouplée à un processus d'évaluation des performances. La société **Mentice**[28] propose, elle aussi, de nombreuses applications pour la simulation de chirurgie minimalement invasive. En particulier leur simulateur d'arthroscopie de l'épaule intègre des modules apprentissage des gestes et de l'anatomie, de diagnostic, et de traitement (Sachdeva et al.,). Dans un autre domaine d'application, **Simedge**[29] propose un simulateur de photo-coagulation laser, PixEyes XL. Cette société développe de nouveaux projets dans les domaines de la gynécologie, de l'obstétrique, de la radiologie et de l'orthopédie. Les travaux de recherches sont effectués en collaboration avec plusieurs universités et instituts de recherche à Lille.

De manière générale, tous les produits que nous venons de présenter sont basés sur des modèles relativement simplifiés, surtout au niveau du comportement biomécanique, afin de garantir une bonne fluidité de la visualisation et du retour d'effort. Parallèlement, de nombreux laboratoires de recherche développent des prototypes de simulateur mettant en œuvre des modèles de déformation plus sophistiqués.

Un projet très ambitieux se développe depuis plusieurs années au sein du groupe de vision par

21. ReachIn : http://www.reachin.se/Medical/
22. Xitact : http://www.xitact.com/
23. ISR : http://dmtwww.epfl.ch/isr/
24. Simbionix : http://www.simbionix.co.il/
25. Boston Dynamics : http://www.bdi.com/Virtual_Surgery.html
26. http://www.bdi.com/Arthroscopy.html
27. http://www.bdi.com/Anastomosis.html
28. Mentice : http://www.mentice.com/
29. Simedge : http://www.simedge.com/

ordinateur de l'ETH de Zürich, dirigé par G. Székely [30]. **LaSSo** [31] est un prototype de simulateur de chirurgie laparoscopique dédié à la chirurgie abdominale et gynécologique (Székely et al., 1998; Székely et al., 2000b; Székely et al., 2000a). Il met en œuvre des modèles déformables d'organes utilisant la loi d'élasticité non-linéaire de Mooney-Rivlin et la méthode des éléments finis sur un maillage hexaédrique. Le schéma d'intégration explicite, ainsi que la non-linéarité du modèle, entraînent une énorme quantité de calculs qui sont pris en charge par une architecture parallèle composée d'un grand nombre de PC et d'une puissante station graphique (Rhomberg et al., 1999). L'interface se compose d'un instrument à retour d'effort *Phantom* [32] dont les degrés de libertés ont été modifiés afin qu'il s'adapte aux mouvements contraints imposés par la laparoscopie (passage par un point fixe). L'utilisateur peut ainsi toucher et saisir les organes virtuels. Par contre, les dernières publications ne faisaient pas état de la prise en compte du retour d'effort. Une autre partie du projet s'attache à mesurer *in vivo* les propriétés biomécaniques des organes, grâce à un outil spécial qui exerce une force de succion sur la surface de l'organe pendant qu'une caméra enregistre la déformation obtenue. Les paramètres du modèle déformable sont ensuite réglés pour qu'il ait le même comportement (Vuskovic et al., 2000).

KISMET Medical Applications [33], dirigé par U.G. Kühnapfel [34] propose une approche un peu différente. Le "Karlsruhe Endoscopic Surgery Trainer" [35] est un prototype de simulateur de chirurgie laparoscopique abdominale (Kuhn et al., 1996; Kühnapfel et al., 1997), qui a été ensuite étendu à la chirurgie gynécologique avec les travaux de thèse de H. K. Çakmak [36] (Çakmak et Kühnapfel, 2000; Kühnapfel et al., 1999). Les modèles déformables sont plus simples que dans le cas précédant, utilisant soit des réseaux masse-ressort, soit l'élasticité linéaire, dans une formulation proche de celle proposée par (Cotin, 1997) et (Bro-Nielsen, 1996), et également proche du modèle pré-calculé que nous présenterons au chapitre 2. Par contre, l'interaction avec les modèles déformables est très complète. Ils utilisent un instrument développé par leurs soins, équipé en retour d'effort, qui permet à l'utilisateur de toucher, saisir, clamper, coaguler et couper (uniquement avec le modèle masse-ressort). S'ajoute à ces gestes un certain nombre d'effets spéciaux : la coagulation s'accompagne d'un dégagement de fumée, les organes peuvent être irrigués, l'utilisateur a la possibilité d'aspirer des liquides, et les battements du cœur se propagent dans les artères (Çakmak et Kühnapfel, 2000). Ils proposent aussi des mesures non invasives des propriétés biomécaniques des tissus vivants grâce à une sonde échographique (Maaß et Kühnapfel, 1999).

Un autre simulateur de chirurgie abdominale et gynécologique en laparoscopie, le **SUSILAP-**

30. G. Székely : http://www.vision.ee.ethz.ch/members/szekely/
31. Laparoscopic Surgery Simulator : http://www.vision.ee.ethz.ch/projects/Lasso/start.html
32. Phantom : http://www.sensable.com/products/phantom.htm
33. KISMET : http://iregt1.iai.fzk.de/KISMET/kis_apps_med.html
34. U.G. Kühnapfel : http://iregt1.iai.fzk.de/KISMET/kismet_people/Kuehnapfel.html
35. Karlsruhe Endoscopic Surgery Trainer : http://iregt1.iai.fzk.de/TRAINER/mic_trainer1.html
36. H. K. Çakmak : http://iregt1.iai.fzk.de/VRTRAIN/phD_main.html

G[37], est développé à l'ISM[38] sous la responsabilité de A. Radetzky[39]. Leur modèle déformable, dénommé "Neuro-Fuzzy" système (Radetzky et al., 1998), est une mise en œuvre d'un modèle masse-ressort faisant usage des systèmes flous et les réseaux de neurones (Kruse et al., 1994; Lin et Lee, 1996; Nauck et al., 1997). Le même laboratoire développe un autre projet sur la simulation de Neurochirurgie, **ROBO-SIM**[40] (Radetzky et al., 2000).

La société **HT Medical**[41] propose un certain nombre de produits développés sur la base des travaux de recherche de M. Bro-Nielsen[42] (Bro-Nielsen, 1996). Ses travaux ont portés sur la simulation de chirurgie générale (application sur la jambe), et plus récemment la chirurgie abdominale ouverte. Deux modèles déformables sont utilisés. Le premier est un modèle volumique, basé sur l'élasticité linéaire et la méthode des éléments finis, dont les calculs sont accélérés par une méthode de condensation et par l'optimisation des structures de données pour les systèmes linéaires creux (Bro-Nielsen et Cotin, 1996). Il utilise aussi des modèles masse-ressort surfaciques et linéiques qu'il peut découper, et sur lesquels il simule le saignement (Bro-Nielsen et al., 1998). Enfin, d'autres travaux ont été menés sur la simulation de neuro-endoscopie (Bro-Nielsen, 1997).

Le groupe VRAI[43], du département de micro-technique de l'Ecole Polytechnique Fédérale de Lausanne (EPFL), a mis au point un simulateur de chirurgie endoscopique, nommé **VIRGY**[44], spécialisé dans la simulation de l'ablation de la vésicule biliaire (Baur et al., 1998). La scène représentant l'abdomen du patient est composée d'objets rigides ou déformables, habillés de textures provenant d'images vidéo. Le modèle de déformation assez simple (surface convoluée avec une fonction donnant le profil de la déformation) est couplé avec un algorithme de découpe. Les différentes tâches qui sont proposées à l'utilisateur sont évaluées afin de rendre compte de ses progrès. Ce simulateur utilise un instrument à retour d'effort spécialisé, le *PantoScope* (Baumann, 1997), développé par l'institut de systèmes robotiques (ISR) de l'EPFL.

Le **MIT Touch Lab**[45] a d'abord proposé des travaux sur la cuisse (Delp et al., 1997), avant de s'intéresser à la chirurgie abdominale en laparoscopie (Basdogan et al., 1998). Le modèle déformable qu'ils utilisent est assez particulier : la déformation locale due à l'instrument est calculée à partir d'une quadrique et les déformations globales sont des *Free Form Deformation* (FFD) (Faloutsos et al., 1997). Appliquer une FFD à un objet correspond à déformer l'espace dans lequel sa position et sa forme sont exprimées. Le simulateur implémente la découpe et le saignement (Basdogan et al., 1999; Delp et al., 1997), et le retour d'effort est appliqué à un *Phantom*. On peut noter, dans la même équipe, des travaux très intéressants sur le rendu haptique pour le toucher de surfaces rugueuses

37. SUrgical SImulator for LAParoscopy in Gynaecology : http://www.ism-austria.at/arne/ara/SUSILAP.html
38. Institute of applied Sciences in Medicine : http://www.ism-austria.at/
39. A. Radetzky : http://www.ism-austria.at/arne/ara/
40. ROBO-SIM : http://www.ism-austria.at/arne/ara/robosim/robosim.html
41. HT Medical : http://www.ht.com/
42. M. Bro-Nielsen : http://www.mortenbronielsen.net/
43. Virtual Reality and Active Interfaces : http://imtsg7.epfl.ch/
44. VIRGY : http://imtsg7.epfl.ch/projects/virgy/
45. MIT Touch Lab : http://touchlab.mit.edu/

(Srinivasan et Basdogan, 1997) ou texturées (Ho et al., 1997; Basdogan et al., 1997).

Le simulateur de coelioscopie gynécologique, **SPIC**[46] (Meseure et Chaillou, 2000; Jambon et al., 2000; Jambon et al., 1997), développé au Laboratoire d'Informatique Fondamentale de Lille (LIFL[47]), repose sur un modèle déformable de type masse-ressort (Meseure et Chaillou, 1997a; Meseure, 1997), couplé à un algorithme de détection de collision optimisé (Meseure et Chaillou, 1997b; Meseure et al., 1998) et à un prototype d'interface à retour d'effort (Lamy et Chaillou, 1999) développée en collaboration avec l'ICAM[48].

Le **MERL**[49] propose plusieurs applications basées sur les travaux de S. Gibson[50]. Leur simulateur de chirurgie arthroscopique du genou (Gibson et al., 1998) repose sur un modèle déformable appelé *3D ChainMail* (Gibson, 1997). C'est un modèle déformable que l'on pourrait qualifier d'hybride. Le calcul de la déformation se fait en deux temps. On applique d'abord un ensemble de contraintes en position sur les éléments : deux éléments voisins n'ont pas le droit de s'éloigner ni de se rapprocher trop l'un de l'autre. Durant cette première étape, le matériau se comporte comme une cote de maille. La seconde étape, plus classique, consiste à minimiser l'énergie élastique du matériau simulé. Le principal avantage de ce modèle est de permettre que les éléments voisins ne s'éloignent pas trop, afin que le problème soit, à chaque itération, très proche de la solution d'équilibre. Une version améliorée de ce modèle permettant de représenter des matériaux non-homogènes est utilisée pour simuler des opérations chirurgicales sur l'œil (Schill et al., 1998). Un *Phantom*, couplé à un algorithme de détection de collision et de modélisation des contacts permet de toucher, couper, déchirer et suturer les organes simulés en ressentant un retour d'effort.

Des simulateurs sont aussi développés pour d'autre organes pour lesquels les interventions sont extrêmement délicates, comme par exemple l'œil. P. Neumann[51] propose un simulateur de vitrectomie (Neumann et al., 1998; Neumann, 2000a; Neumann, 2000b) reposant sur un très beau modèle géométrique de l'œil. Le comportement biomécanique est approché par plusieurs couches de réseaux masse-ressort volumiques et surfaciques. Des instruments à positionnement magnétique permettent d'interagir avec le modèle et d'effectuer plusieurs gestes chirurgicaux. Un effort particulier a été fait pour l'évaluation et la validation de ce simulateur en milieu hospitalier. Des travaux très proches ont été menés par M. Schill[52], mais en utilisant cette fois le modèle déformable 3D ChainMail amélioré (Schill et al., 1998), développé sur la base des travaux de S. Gibson.

D'autres laboratoires proposent des simulateurs souvent relativement spécialisés. Le **National Biocomputation Center**[53] (Stanford/Nasa) développe plusieurs applications pour la simulation de chirurgie gynécologique, de microchirurgie, ou encore de rééducation avec des environnements

46. SPIC : http://www.lifl.fr/GRAPHIX/rechappli/simulateur/spic/index.html
47. LIFL : http://www.lifl.fr/
48. Institut Catholique d'Arts et Métiers : http://www.icam.fr/v2/index2.htm
49. Mitsubishi Electric Research Laboratory : http://www.merl.com/
50. S. Gibson : http://www.merl.com/people/gibson/
51. P. Neumann : http://www.sbhis.uic.edu/paul/
52. M. Schill : http://www-li5.ti.uni-mannheim.de/~schill/www/
53. National Biocomputation Center : http://www-biocomp.stanford.edu/

virtuels et des interfaces haptiques. **Georgia Tech**[54] travaille sur un simulateur de chirurgie spécialisé sur les vaisseaux et autres structures tubulaires (Cover et al., 1993). Il faut aussi noter les travaux menés au **HIT Lab**[55] par J. Berkley[56] sur la simulation de suture. Il reconstruit des modèles éléments finis élastiques linéaires à partir d'images médicales (Berkley et al., 2000), qu'il utilise en temps-réel grâce à des pré-calculs pour simuler les gestes de suture grâce à deux interfaces *Phantom* (Berkley et al., 1999). Au sein du même laboratoire sont également développées des interfaces immersives pour la simulation de chirurgie (Oppenheimer et Weghorts, 1999).

5.3 Les projets AISIM et CAESARE

L'élaboration d'un simulateur de chirurgie soulève un grand nombre de problèmes récurrents dans les domaines de la vision par ordinateur et de l'animation (Ayache et al., 1998; Marescaux et al., 1998). C'est pour traiter le sujet dans sa globalité que l'INRIA a décidé de promouvoir les projets AISIM et CAESARE dont nous avons parlé au début de ce chapitre. Ces projets s'appuient sur les travaux effectués par Stéphane Cotin durant sa thèse au sein de l'équipe Epidaure (Cotin, 1997). Il a développé deux modèles déformables, basés sur l'élasticité linéaire et la méthode des éléments finis, le modèle quasi-statique pré-calculé (Cotin et al., 1999), très efficace mais qui ne supporte pas les changements de topologie, et le modèle dynamique masse-tenseur (Delingette et al., 1999), plus lent mais qui permet de simuler les découpes. Enfin, il a proposé une méthode pour coupler les deux modèles afin d'en additionner les avantages au sein d'un modèle hybride (Cotin et al., 2000). Ces modèles déformables prenaient place dans un simulateur de chirurgie hépatique en laparoscopie avec retour d'effort.

Une première composante des projets AISIM et CAESARE est la construction d'un modèle déformable du foie "sans compromis", c'est-à-dire en particulier sans la contrainte de devoir calculer les déformations en temps-réel. Ce modèle pourra ainsi servir de référence pour évaluer le degré de réalisme des autres modèles déformables qui ont du être simplifiés afin de fonctionner en temps-réel. Pour cela M. Vidrascu (équipe MACS) a développé un modèle déformable du foie incompressible, utilisant la loi d'élasticité non-linéaire de Mooney-Rivlin, couplé à une coque externe qui représente la capsule de Glisson. L'utilisation d'un algorithme de décomposition de domaine a permis de réduire les temps de calcul de plusieurs heures à quelques minutes (Vidrascu, 1999). Parallèlement, M. Thiriet (équipe M3N) s'est intéressé à l'étude anatomique du foie. Il est en effet important de connaître les relations physiques et anatomiques entre le foie et les organes voisins. La seconde partie de son travail consiste à déterminer les coefficients d'élasticité caractérisant le comportement biomécanique du foie, en particulier le module de Young. Il a, pour cela, collaboré avec D. Dan qui effectuait une thèse sur le comportement mécanique du foie dans des situations de choc (Dan et Flaud, 1998; Dan, 1999; Dan et al., 2000).

Le second grand thème abordé est celui des modèles déformables fonctionnant en temps-réel. Ce

54. Georgia Tech : http://www.cc.gatech.edu/gvu/biovis/surgsim/
55. Human Interface Technologie Lab : http://www.hitl.washington.edu/
56. J. Berkley : http://www.hitl.washington.edu/people/jberkley/

sujet est traité par trois équipes qui ont suivi des directions différentes. Au sein de l'équipe SHARP, le choix s'est tourné vers les réseaux masse-ressort. D. d'aulignac et F. Boux de Casson proposent des modèles masse-ressort mettant en œuvre différents types de ressorts et de comportements adaptés aux particularités anatomiques des structures qu'ils représentent (peau, muscle, parenchyme, ...) (d'Aulignac et al., 1999a; d'Aulignac et al., 1999b; Boux de Casson et Laugier, 1999). D. d'aulignac a, d'autre part, beaucoup travaillé sur les problèmes liés au couplage d'un modèle déformable avec une interface à retour d'effort (d'Aulignac et Balaniuk, 1999; d'Aulignac et al., 2000). De plus, F. Boux de Casson s'est intéressé à la simulation du déchirement d'un modèle déformable lorsqu'il est soumis à des contraintes trop importantes (Boux de Casson et Laugier, 2000; Boux de Casson, 2000). G. Debunne (équipe iMAGIS) s'est orienté vers un modèle utilisant l'élasticité linéaire et la méthode des volumes finis. L'originalité de son approche est de pouvoir calculer les déformations à plusieurs résolutions, ce qui permet d'augmenter la précision dans les zones où la déformation est importante et de gagner du temps sur le reste du modèle en y diminuant la résolution. Il a d'abord appliqué ce principe sur des grilles régulières (maillages hexaédriques) que se prêtent bien aux subdivisions locales (Debunne et al., 1999), puis s'est tourné vers les maillages tétraédriques (Debunne et al., 2000). Les travaux réalisés au sein de l'équipe Epidaure seront présentés en détail dans ce manuscrit. Nous avons développé des modèles déformables basés sur la théorie de l'élasticité et la méthode des éléments finis, et nous nous sommes particulièrement intéressés à la modélisation des matériaux linéaires anisotropes (Picinbono et al., 2000c), puis non-linéaires (Picinbono et al., 2000b; Picinbono et al., 2000a; Picinbono et al., 2001a). Nous avons aussi travaillé sur la modélisation des contacts entre les instruments chirurgicaux et les organes (Picinbono et al., 2001b). Enfin, nous proposons une solution au problème du couplage des modèles déformables avec des systèmes à retour d'effort (Picinbono et Lombardo, 1999).

D'autres sujets que l'on rencontre souvent dans le domaine de l'animation par ordinateur ont été traités, donnant des résultats particulièrement intéressants. C'est le cas notamment pour la détection de collision entre un objet rigide (l'outil du chirurgien) et un modèle déformable (l'organe simulé), avec un algorithme original faisant appel au processus de rendu OpenGL (Lombardo et al., 1999), que nous utilisons dans nos travaux et qui sera présenté au chapitre 4. Une dernière contribution a été proposée par F. Neyret (iMAGIS) pour résoudre les problèmes qui se posent lorsque l'on désire texturer la surface d'un objet à la géométrie complexe et irrégulière (Neyret et Cani, 1999). Nous présentons une application de ces travaux au chapitre 2.

6 Contributions

Ce manuscrit de thèse se divise en 5 chapitres. Le chapitre 1 pose les bases mathématiques et physiques nécessaires à la bonne compréhension des chapitres suivants. Ainsi, nous commençons par y introduire la mécanique des milieux continus, théorie de l'élasticité et les lois des systèmes dynamiques. Nous poursuivons par un exposé rapide de la méthode des éléments finis et de sa formulation particulière pour les maillages tétraédriques. La seconde partie de ce chapitre détaille les notations de calcul matriciel et différentiel que nous utilisons.

Les chapitres 2 et 3 sont consacrés aux travaux que nous avons menés sur les modèles déformables. Dans le chapitre 2, nous commençons par une description détaillée du modèle d'élasticité linéaire. Nous lui associons deux méthodes de résolution donnant naissance à deux modèles déformables : un modèle quasi-statique pré-calculé, très efficace mais dont on ne peut pas modifier la topologie, et un modèle inspiré des réseaux masse-ressort, qui est dynamique et qui se prête particulièrement bien à la simulation de la découpe, le modèle masse-tenseur. Nous proposons ensuite une généralisation de ces modèles déformables aux matériaux dits **transversalement isotropes**, c'est-à-dire les matériaux qui possèdent une direction de déformation privilégiée. Ce nouveau modèle nous permet de simuler les déformations de matériaux fortement anisotropes comme les muscles, les tendons ou les vaisseaux sanguins. Nous sommes aussi capables de construire des modèles comportant des zones isotropes et d'autres anisotropes. Cela nous permet notamment de représenter le parenchyme hépatique qui est isotrope mais qui est traversé par des vaisseaux qui le rendent localement anisotrope. Pour compléter le réalisme biomécanique de notre modèle de foie, nous proposons de coupler le modèle masse-tenseur à un **modèle masse-ressort surfacique** afin de modéliser la raideur plus importante de la surface du foie due à la présence d'une peau entourant l'organe : la capsule de Glisson.

Malgré les améliorations que nous y avons apportées, le modèle d'élasticité linéaire reste limité par le fait qu'il n'est valable que pour les petits déplacements. En effet, l'absence d'invariance par rotation provoque des déformations irréalistes dès que le déplacement d'une partie du modèle devient trop grande. Nous proposons donc, dans le chapitre 3, un nouveau modèle masse-tenseur **non-linéaire**, basé sur l'utilisation du modèle d'élasticité de **St Venant-Kirchhoff** , qui est l'équivalent de l'élasticité linéaire, mais qui reste valable pour les **grands déplacements**. De plus, afin de mettre à profit le réalisme des déformations qu'il permet de simuler, nous le généralisons lui aussi aux matériaux anisotropes en suivant le même cheminement que pour le modèle linéaire. Ce modèle non-linéaire étant plus coûteux en temps de calcul, nous proposons un **modèle adaptatif**, qui utilise l'élasticité non-linéaire seulement dans les zones où le déplacement est important, le reste du modèle se contentant d'un comportement linéaire. Enfin, nous proposons une nouvelle formulation de l'énergie élastique de St Venant Kirchhoff, que nous exprimons uniquement en fonction des variations de longueur des arêtes du maillages. Nous développons ce modèle, que nous appelons **simplexe hyperélastique**, dans les cas 2D (maillage surfacique triangulaire) et 3D (maillage volumique tétraédrique). Cette formulation dans laquelle l'invariance par rotation est intrinsèque, permet d'améliorer la stabilité et l'efficacité du modèle déformable.

Dans le chapitre 4, nous présentons les travaux que nous avons menés pour intégrer les modèles déformables que nous avons développés au sein d'un simulateur de chirurgie. Nous proposons tout d'abord une **modélisation des contacts** et des interactions entre l'organe virtuel et les instruments des chirurgiens. Nous définissons des comportements élémentaires permettant à l'outil de glisser sur la surface du foie, de la saisir au moyen de la mâchoire d'une pince mais aussi de découper le parenchyme hépatique. Nous abordons ensuite le couplage proprement dit du modèle déformable avec une interface mécanique à retour d'effort. Ce système à retour d'effort étant relié à un ordinateur différent de celui qui fait fonctionner le modèle déformable, nous décrivons le **protocole de communication** que nous avons mis en place afin que les deux ordinateurs échangent des informations de position et de force. Finalement, nous proposons une solution originale au problème du **retour d'effort**, permettant d'obtenir, grâce à une méthode d'**extrapolation des forces**, un rendu haptique réaliste en se contentant de calculer les forces à la fréquence imposée par le rendu visuel en temps-réel.

Le chapitre 5 propose un condensé des travaux présentés dans les chapitres précédents sous la forme d'une description chronologique des étapes successives nécessaires à la construction d'un simulateur de chirurgie. En partant de la construction du modèle géométrique de l'organe à partir d'images médicales, nous poursuivons par l'intégration des connaissances anatomiques et rhéologiques du foie et des conditions dans lesquelles se déroulent les opérations. Finalement, nous présentons des exemples de simulation de gestes chirurgicaux, rendus possible par l'intégration des différents travaux réalisés durant cette thèse.

Enfin, le dernier chapitre conclut ces travaux et propose un certain nombre d'orientations pour les développements futurs sur le sujet de la simulation de chirurgie.

Chapitre 1

Les bases mathématiques et physiques

Dans ce premier chapitre, nous introduirons les notations et les concepts mathématiques et physiques de base que nous utiliserons tout au long du manuscrit. Nous y décrirons les principes de la mécanique des milieux continus et la théorie de l'élasticité, et nous poursuivrons par une brève présentation de la méthode des éléments finis. Nous aborderons ensuite les notions de calcul vectoriel, matriciel et différentiel.

« Les mathématiques sont la seule science où l'on ne sait pas de quoi l'on parle, ni si ce qu'on dit est vrai »
Russell

1.1 Introduction

Ce chapitre rassemble toutes les notations et tous les concepts de base nécessaires à la compréhension du reste du manuscrit. Sans entrer dans les détails, nous donnerons un aperçu le plus synthétique possible des différents outils mathématiques et physiques que nous utiliserons, et nous orienterons le lecteur vers les références bibliographiques qui nous ont servi de références ou qui nous semblent pertinentes pour une étude plus approfondie.

Nous commencerons par aborder les thèmes de la physique des milieux continus et de l'élasticité, pour pourvuivre avec une présentation de la méthode des éléments finis. Les deux derniers paragraphes s'attacheront à introduire les bases des calculs vectoriels, matriciels et différentiels qui sont utilisés tout au long du document.

1.2 La physique des milieux continus et des systèmes dynamiques

La pièce maîtresse d'un simulateur de chirurgie est le modèle déformable qui représente l'organe et les structures anatomiques avec lesquels il va falloir interagir. Afin d'obtenir un comportement réaliste de notre organe virtuel, nous avons choisi de construire nos modèles déformables sur les bases de la mécanique des milieux continus.

La mécanique des milieux continus permet de décrire le comportement d'un corps qui se déforme ou se déplace sous l'influence de contraintes externes. Dans le cas d'un corps "mou", nous pourrons alors exprimer et quantifier sa déformation lorsqu'il est soumis à un champ de forces volumiques comme la pesanteur ou bien lorsque des forces sont appliquées sur sa surface. Cette déformation, ainsi que les contraintes qu'elle engendre à l'intérieur du matériau, vont permettre de calculer le mouvement résultant du corps.

Les caractéristiques mécaniques d'un matériau sont traduites par la relation qui existe entre la déformation subie par le corps et les contraintes internes qui en résultent. Nous nous placerons alors dans le cas du modèle d'élasticité de St Venant-Kirchhoff, qui est un cas particulier des matériaux hyperélastiques.

Pour plus de détails sur les concepts abordés dans ce paragraphe, nous conseillons les ouvrages de D. François (François et al., 1990), de Y. C. Fung (Fung, 1965) et de A. J. M. Spencer (Spencer, 1984), ainsi que les cours de J. Garrigues (Garrigues, 1999d) et de T.G. Shawki (Shawki, 1996) sur la mécanique des milieux continus. En ce qui concerne le théorie de l'élasticité, le cours très pédagogique de J. Garrigues (Garrigues, 1999a) pourra être approfondi par l'étude des ouvrages de P.G. Ciarlet (Ciarlet, 1988; Ciarlet, 1990; Ciarlet, 1997), de J.T. Oden (Oden, 1967) et de R.W. Ogden (Ogden, 1984).

1.2.1 La déformation

Considérons Ω, un ouvert de $\mathbb{R}^3$, $\bar{\Omega}$ son adhérence et Γ sa frontière. Ce domaine passe de sa position de repos à sa position déformée sous l'action de la déformation $\mathbf{\Phi}$. En tout point $\mathbf{X}$ du domaine, on peut représenter de manière équivalente la déformation par le champ de déplacements $\mathbf{U}(\mathbf{X})$ qu'elle induit (figure 1.1) :

$$\mathbf{\Phi}(\mathbf{X}) = \mathbf{X} + \mathbf{U}(\mathbf{X}). \tag{1.1}$$

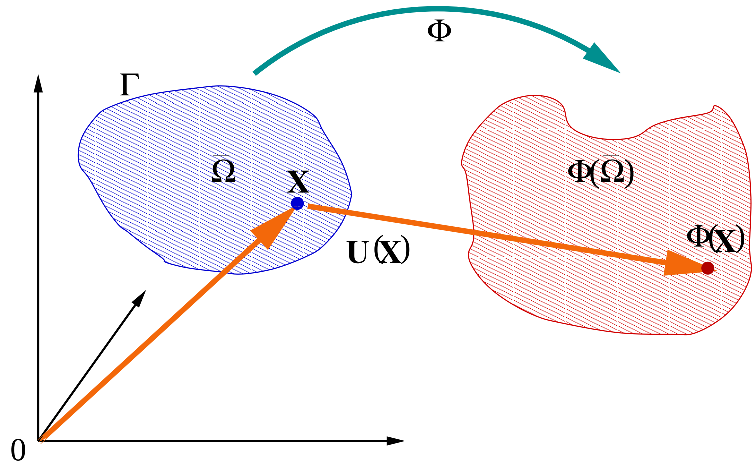


Fig. 1.1 – *Application d'une déformation sur un domaine de* $\mathbb{R}^3$

Afin de quantifier la déformation d'un élément de volume du domaine, on définit un tenseur de déformation, c'est-à-dire une matrice 3x3 symétrique qui dépend du gradient de la déformée. Lorsque l'on raisonne dans un référentiel Lagrangien (la configuration de repos sert de référence), on utilise le tenseur de déformation de Cauchy-Green C (Maurel et al., 1998; Ciarlet, 1988), défini par :

$$C = \nabla\mathbf{\Phi}^t \, \nabla\mathbf{\Phi}. \tag{1.2}$$

Pour que ce tenseur permette de caractériser la déformation subie par un élément de volume, il se doit d'être invariant par transformation rigide, c'est-à-dire par translation et par rotation. On vérifie en effet que pour une transformation rigide quelconque $\mathbf{\Phi}_{rig}(\mathbf{X}) = T + R\mathbf{X}$ (où T est une translation et R une rotation), C est bien égal à la matrice identité :

$$\nabla\mathbf{\Phi}_{rig} = R \tag{1.3}$$

$$C = \nabla\mathbf{\Phi}_{rig}^t \, \nabla\mathbf{\Phi}_{rig} = R^t \, R = I \tag{1.4}$$

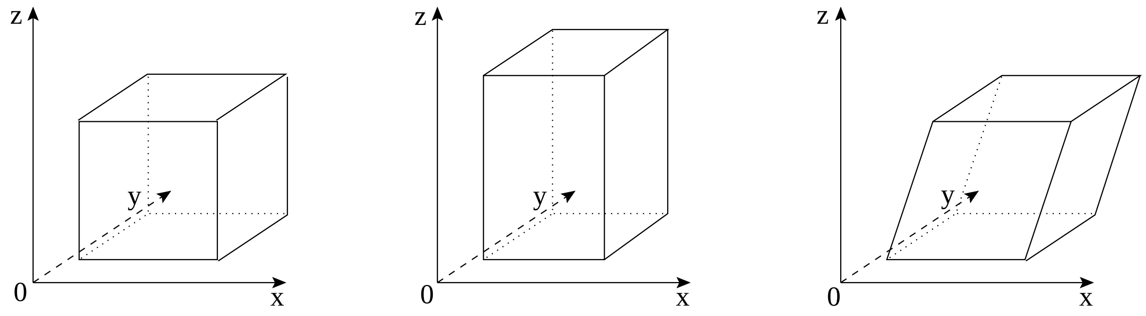
Ce tenseur ne tient donc compte que de la partie non rigide de la déformation. Si l'on considère un élément de volume cubique $dV = dx \; dy \; dz$, C exprime sa variation de forme. On peut distinguer deux types de changement de forme :

- Les trois éléments diagonaux expriment l'étirement du cube le long de ses trois axes (figure 1.2(b)). Par exemple l'étirement (ou la contraction) dans la direction z est quantifié par :

$$C_{22} = \left(\frac{\partial\phi}{\partial z}\right)^2 + \left(\frac{\partial\psi}{\partial z}\right)^2 + \left(\frac{\partial\zeta}{\partial z}\right)^2 = \left(\frac{\partial\mathbf{\Phi}}{\partial z}\right) \cdot \left(\frac{\partial\mathbf{\Phi}}{\partial z}\right) = \left\|\frac{\partial\mathbf{\Phi}}{\partial z}\right\|^2 \tag{1.5}$$

- Les trois autres éléments (c'est une matrice symétrique) expriment le cisaillement, c'est-à-dire la variation d'angle entre les faces du cube (figure 1.2(c)). Pour l'angle entre les faces $x = C^{ste}$ et $z = C^{ste}$, on a :

$$C_{02} = \frac{\partial \phi}{\partial x}\frac{\partial \phi}{\partial z} + \frac{\partial \psi}{\partial x}\frac{\partial \psi}{\partial z} + \frac{\partial \zeta}{\partial x}\frac{\partial \zeta}{\partial z} = \frac{\partial \mathbf{\Phi}}{\partial x}.\frac{\partial \mathbf{\Phi}}{\partial z} \tag{1.6}$$



(a) Forme au repos (b) Étirement (c) Cisaillement

FIG. 1.2 – *Les composantes de la déformation d'un élément de volume cubique*

Comme le montre la figure 1.1, on peut raisonner de manière équivalente sur le champ de déplacements. On remarque alors que le gradient du champ de déplacements est relié à celui du champ de déformations par la relation $\nabla\mathbf{\Phi} = I + \nabla\mathbf{U}$ (I est la matrice identité). On utilisera dans ce cas le tenseur de déformation de Green-St Venant E, qui s'exprime en fonction du gradient du champ de déplacements :

$$E = \frac{1}{2}(C - I) = \frac{1}{2}\left(\nabla\mathbf{U}^t + \nabla\mathbf{U} + \nabla\mathbf{U}^t\nabla\mathbf{U}\right) \tag{1.7}$$

Comme pour le tenseur C, on peut différencier les trois éléments diagonaux de E, que l'on notera ϵ_x, ϵ_y et ϵ_z. Ils représentent l'étirement de l'élément de volume dans les trois directions de l'espace. Les trois autres éléments, γ_{xy}, γ_{xz} et γ_{yz}, traduisent le cisaillement dans l'élément de volume. Le tenseur E s'exprime alors en fonction de ces six composantes de déformation sous la forme :

$$E = \frac{1}{2}\begin{bmatrix} 2\,\epsilon_x & \gamma_{xy} & \gamma_{xz} \\ \gamma_{xy} & 2\,\epsilon_y & \gamma_{yz} \\ \gamma_{xz} & \gamma_{yz} & 2\,\epsilon_z \end{bmatrix} \quad \text{où} \quad \begin{cases} \epsilon_x = u_x + \frac{1}{2}(u_x^2 + v_x^2 + w_x^2) \\ \epsilon_y = v_y + \frac{1}{2}(u_y^2 + v_y^2 + w_y^2) \\ \epsilon_z = w_z + \frac{1}{2}(u_z^2 + v_z^2 + w_z^2) \\ \gamma_{xy} = (u_y + v_x) + (u_x u_y + v_x v_y + w_x w_y) \\ \gamma_{xz} = (u_z + w_x) + (u_x u_z + v_x v_z + w_x w_z) \\ \gamma_{yz} = (v_z + w_y) + (u_y u_z + v_y v_z + w_y w_z) \end{cases} \tag{1.8}$$

Les modèles déformables étudiés dans le chapitre 2 reposent sur une loi de déformation élastique linéaire. Dans ce cas particulier de la mécanique des milieux continus, on ne considère que la partie linéaire du tenseur E, qui prend alors la forme :

$$E_l = \frac{1}{2}\left(\nabla\mathbf{U}^t + \nabla\mathbf{U}\right) \tag{1.9}$$

Les six composantes de déformation valent alors :

$$\begin{cases} \epsilon_x^l = u_x & \epsilon_y^l = v_y & \epsilon_z^l = w_z \\ \gamma_{xy}^l = (u_y + v_x) & \gamma_{xz}^l = (u_z + w_x) & \gamma_{yz}^l = (v_z + w_y) \end{cases} \tag{1.10}$$

1.2.2 Les forces

La déformation du domaine Ω que nous considérons peut résulter de l'application de deux types de forces :

- Les forces volumiques appliquées sur une partie ou sur la totalité du domaine. L'exemple le plus simple est la force de gravité. On notera $\mathbf{f}$ la densité de l'ensemble de ces forces :

$$\mathbf{f} : \bar{\Omega} \longrightarrow \mathbb{R}^3 \tag{1.11}$$

- Les forces surfaciques appliquées sur la partie libre de la frontière du domaine Γ_1, le reste Γ_0 de la frontière du domaine ($\Gamma = \Gamma_0 \cup \Gamma_1$) étant contrainte en déplacements. Ce sont par exemple des forces de pression ou de contact. La densité de ces forces surfaciques sera alors notée :

$$\mathbf{g} : \Gamma_1 \longrightarrow \mathbb{R}^3 \tag{1.12}$$

1.2.3 Les contraintes

À partir des axiomes de conservation locale du moment linéaire et du moment angulaire, et en appliquant le *Théorème de Cauchy* (Ciarlet, 1988), on montre que pour tout élément de volume à l'intérieur du corps considéré, il existe un champ de vecteurs $\tilde{\mathbf{t}}(\tilde{\mathbf{X}},\tilde{\mathbf{n}})$ et un champ de tenseurs $\tilde{T}(\tilde{\mathbf{X}})$ tels que :

$$\tilde{\mathbf{t}}(\tilde{\mathbf{X}},\tilde{\mathbf{n}}) = \tilde{T}(\tilde{\mathbf{X}})\tilde{\mathbf{n}}, \tag{1.13}$$

où $\tilde{\mathbf{n}}$ est le vecteur unitaire normal à la surface de l'élément de volume. $\tilde{\mathbf{t}}$ et $\tilde{T}$ sont respectivement le *vecteur de contrainte de Cauchy* et le *tenseur de contrainte de Cauchy*. Le *"tilde"* sur le nom des variables signifie qu'elles sont exprimées dans le référentiel **Eulérien**, où la position déformée est prise comme référence. Afin de pouvoir raisonner dans le référentiel **Lagrangien** (la position au repos est prise comme référence), on utilise la transformée de Piola pour construire le *premier tenseur de contrainte de Piola-Kirchhoff* :

$$T(\mathbf{X}) = det(\nabla\mathbf{\Phi}(\mathbf{X}))\,\tilde{T}(\tilde{\mathbf{X}})\,\left(\nabla\mathbf{\Phi}(\mathbf{X})^t\right)^{-1}, \quad \tilde{\mathbf{X}} = \mathbf{\Phi}(\mathbf{X}). \tag{1.14}$$

Cela permet naturellement de définir le *premier vecteur de contraintes de Piola-Kirchhoff* $\mathbf{t}$, de telle manière que l'équation :

$$\mathbf{t}(\mathbf{X},\mathbf{n}) = T(\mathbf{X})\mathbf{n} \tag{1.15}$$

soit toujours vérifiée. Par contre ce tenseur de Piola-Kirchhoff n'est pas symétrique alors que le tenseur de Cauchy l'était. Comme il est préférable de considérer un tenseur de contrainte qui soit symétrique, en particulier parce que dans ce cas les équations constitutives prennent une forme plus simple, nous utilisons le *second tenseur de contrainte de Piola-Kirchhoff* Σ :

$$\Sigma(\mathbf{X}) = \nabla\mathbf{\Phi}(\mathbf{X})^{-1}\, T(\mathbf{X}). \tag{1.16}$$

1.2.4 Le mouvement

En écrivant l'axiome de conservation globale du moment linéaire sur tout sous-domaine de Ω, et en appliquant le *théorème de divergence de Gauss*, le *théorème de transport* et l'*équation de continuité*, on obtient l'équation générale du mouvement pour un milieu continu (chapitre 4 de (Bro-Nielsen, 1996)). Pour un mouvement élastique, cette équation correspond à un *mouvement harmonique*, à laquelle on rajoute en général un terme d'amortissement pour obtenir l'équation du *mouvement harmonique amorti*, que l'on appelle aussi équation Lagrangienne du mouvement :

$$\begin{aligned} \rho\frac{d^2\mathbf{U}}{dt^2} + c\frac{d\mathbf{U}}{dt} &= \mathbf{f}(\mathbf{X}) + \nabla.T(\mathbf{X}), & \mathbf{X} \in \Omega \qquad (1.17)\\ \mathbf{g}(\mathbf{X}) &= T(\mathbf{X})\mathbf{n}, & \mathbf{X} \in \Gamma_1 \end{aligned}$$

$$\text{où} \begin{cases} \rho & \text{est la densité de masse} \\ c & \text{est le coéfficient d'amortissement du matériau} \\ \mathbf{f} & \text{est la densité de force volumique} \\ \mathbf{g} & \text{est la densité de force surfacique} \end{cases}$$

Pour une déformation statique, l'équation 1.17 devient :

$$0 = \mathbf{f}(\mathbf{X}) + \nabla.T(\mathbf{X}) \tag{1.19}$$

1.2.5 Les matériaux hyperélastiques

Depuis le début de ce paragraphe 1.2, nous avons vu, d'une part, comment quantifier une déformation (équations 1.2, 1.7 et 1.8), et d'autre part comment le tenseur de contrainte gouverne les équations du mouvement (équation 1.17). Il ne manque plus alors qu'une relation liant les contraintes et les déformations pour établir le comportement biomécanique du matériau considéré. Pour cela, nous nous plaçons dans le cadre des matériaux hyperélastiques.

Un matériau est dit **élastique**, si le tenseur de contraintes qui le caractérise dépend uniquement de la position et du gradient de la déformation. De plus, ce même matériau sera **hyperélastique** s'il existe une fonction d'énergie potentielle W telle que le tenseur de contraintes dérive de cette énergie :

$$\Sigma = \nabla\mathbf{\Phi}^{-1}\, T = \frac{\partial W}{\partial E}. \tag{1.20}$$

On définira alors l'énergie totale de déformation sur tout le domaine Ω à partir de l'énergie potentielle :

$$\mathcal{E} = \int_\Omega W \; d\mathbf{X} \tag{1.21}$$

Pour les matériaux isotropes et homogènes, la fonction d'énergie potentielle W est complètement caractérisée par les invariants principaux du tenseur de déformation de Cauchy-Green C. Ces invariants sont les coefficients du polynôme caractéristique $det(C - \lambda I)$, et s'écrivent en fonction de C ou du tenseur de déformation de Green-St Venant E :

$$\begin{cases} l_1 = & trC & = 3 + 2trE \\ l_2 = & \frac{1}{2}\left((trC)^2 - trC^2\right) & = 3 + 4trE + 2\left[(trE)^2 - trE^2\right] \\ l_3 = & |C| & = |2E + I| \end{cases} \tag{1.22}$$

À l'aide de ces invariants, on peut exprimer la fonction d'énergie potentielle pour un matériau hyperélastique quelconque comme une somme infinie de la forme :

$$W = \sum_{r,s,t=0}^{\infty} L_{rst}(l_1 - 3)^r(l_2 - 3)^s(l_3 - 1)^t, \quad L_{000} = 0 \tag{1.23}$$

Il est intéressant de noter que le troisième invariant mesure la variation locale de volume. Pour des matériaux incompressibles ce terme vaut 1. Une approche souvent utilisée pour renforcer la contrainte d'incompressibilité est d'introduire un multiplicateur de Lagrange $\mathcal{L}$ qui se comporte comme une pression interne obligeant le matériau à trouver un état d'équilibre garantissant son incompressibilité. La version incompressible de la fonction d'énergie potentielle peut ainsi s'écrire :

$$W^i = \sum_{r,s=0}^{\infty} L^i_{rs}(l_1 - 3)^r(l_2 - 3)^s + \mathcal{L}(l_3 - 1), \quad L^i_{00} = 0 \tag{1.24}$$

Dans les deux cas, la donnée de l'ensemble des constantes L_{rst} (ou L^i_{sr} pour les matériaux incompressibles) permet de définir un modèle de comportement et donc un modèle de matériau hyperélastique.

1.2.6 Le modèle d'élasticité de St Venant-Kirchhoff

Le modèle d'élasticité de St Venant-Kirchhoff est défini à l'aide de l'équation 1.23 en choisissant les constantes L_{rst} toutes nulles sauf :

$$L_{100} = \mu, \quad L_{200} = \frac{\lambda + 2\mu}{8} \quad \text{et} \quad L_{010} = -\frac{\mu}{2} \tag{1.25}$$

En utilisant les expressions des invariants principaux en fonction de E (équations 1.22), on obtient la formulation de l'énergie potentielle élastique du matériau de St Venant-Kirchhoff :

$$W = \frac{\lambda}{2}(trE)^2 + \mu\, trE^2, \tag{1.26}$$

où λ et μ sont les constantes de *Lamé*. Elles sont reliées aux grandeurs classiques exprimant les propriétés des matériaux élastiques, c'est-à-dire le **module de Young** E et le **rapport de Poisson** ν, par les formules :

$$\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)} \qquad \mu = \frac{E}{2(1+\nu)} \tag{1.27}$$

Le module de Young E caractérise la raideur du matériau, tandis que le rapport de Poisson ν représente son incompressibilité. Il est à noter que ce matériau est physiquement linéaire, puisque la relation entre contraintes et déformations est linéaire :

$$\Sigma = \frac{\partial W}{\partial E} = \lambda \, (trE) \, I + 2\mu \, E. \tag{1.28}$$

Par contre, la relation non-linéaire entre E et $\nabla \mathbf{U}$ conduit à une non linéarité géométrique. On parle aussi, dans ce cas, d'élasticité en grands déplacements.

Les équations 1.26 et 1.28 effectuent la jonction entre contraintes et déformations, ce qui permet de totalement caractériser le comportement biomécanique d'un matériau élastique obéissant au modèle de St Venant-Kirchhoff. Le tableau suivant propose une récapitulatif du processus de calcul de la déformation pour ce modèle d'élasticité : la déformation est quantifiée par le tenseur déformation. Les invariants principaux de ce tenseur et les coefficients de Lamé permettent d'exprimer l'énergie élastique, dont on dérive le tenseur des contraintes. Ces contraintes internes, couplées aux contraintes volumiques et surfaciques externes appliquées sur le domaines, permettent de déduire l'équation du mouvement du modèle.

$\mathbf{\Phi}(\mathbf{X}) = \mathbf{X} + \mathbf{U}$

$C = \nabla \mathbf{\Phi}^t \nabla \mathbf{\Phi}$

$E = \frac{1}{2}[\nabla \mathbf{U}^t + \nabla \mathbf{U} + \nabla \mathbf{U}^t \nabla \mathbf{U}]$

Déplacements —— **Tenseur de déformation :** *Strain*

Énergie Potentielle de Déformation :
$W = \frac{\lambda}{2}(trE)^2 + \mu tr E^2$

Second Tenseur de Contrainte
de Piola-Kirchoff :
$\Sigma = \lambda(trE)I + 2\mu E$

Forces —— **Tenseur de contrainte :** *Stress*

Équation du mouvement :
$\rho \frac{d^2\mathbf{U}}{dt^2} + c\frac{d\mathbf{U}}{dt} = \mathbf{f}(\mathbf{X}) + \nabla . T(\mathbf{X})$

Premier Tenseur de Contrainte
de Piola-Kirchoff T :
$\Sigma(\mathbf{\Phi},\mathbf{X}) = \nabla \mathbf{\Phi}(\mathbf{X})^{-1} T(\mathbf{X}) = \frac{\partial W}{\partial E}$

1.3 La méthode des éléments finis

Le but de ce paragraphe est de familiariser le lecteur avec les principes de la méthode des éléments finis qui sera mise en œuvre dans les chapitres 2 et 3. Nous n'en présenterons donc que les grandes lignes et nous nous concentrerons sur l'étude des éléments finis tétraédriques de type P_1 que nous utiliserons par la suite. Pour une étude plus approfondie, nous conseillons les cours de J. Garrigues (Garrigues, 1999b) et de P.J. Hunter (Hunter et Pullan, 1998), ainsi que le chapitre 2 de la thèse de S. Cotin (Cotin, 1997), ou bien encore le chapitre 3 du livre de W. Maurel *et al.* (Maurel et al., 1998) et les ouvrages de référence de O. Zienkiewicz (Zienkiewicz, 1977), de K.-L. Bathe (Bathe, 1982) et de T. Hughes (Hughes, 1987).

Si les modèles physiques choisis pour modéliser un problème réel nous fournissent en général les équations et les conditions aux limites, les mathématiques, par contre, sont souvent impuissantes à nous fournir une solution analytique de ce problème. Heureusement, l'analyse numérique vient à notre secours pour apporter des solutions numériques par l'intermédiaire de plusieurs méthodes, parmi lesquelles on peut citer la méthode des différences finies, la méthode des volumes finis, et la méthode des éléments finis à laquelle nous allons nous intéresser maintenant.

La plupart des problèmes de physique peuvent se formuler sous la forme : "trouver un champ (scalaire, vectoriel ou tensoriel) $\mathbf{H}(\mathbf{X},t)$ satisfaisant à un ensemble d'équations pour tous les points $\mathbf{X}$ du domaine Ω et à tout instant t, et respectant des conditions aux limites sur la frontière Γ du domaine".

La méthode des éléments finis consiste alors à rechercher une solution approchée de la solution exacte sous la forme d'un champ $\tilde{\mathbf{H}}(\mathbf{X},t)$ défini par morceaux sur des sous-domaines de Ω. Les n sous-domaines Ω_i doivent représenter le domaine Ω tout entier et doivent être disjoints deux à deux, c'est-à-dire qu'ils doivent vérifier :

$$\bigcup_{i=0}^{n-1} \Omega_i = \Omega \qquad \text{et} \qquad \Omega_i \cap \Omega_j = \emptyset,\ \ \forall i \neq j \tag{1.29}$$

Les champs $\tilde{\mathbf{h}}_i(\mathbf{X},t)$ définis sur les sous-domaines, sont choisis parmi une famille de champs arbitraires (généralement polynomiaux). La famille de champs locaux est appelée *espace des fonctions d'interpolation de l'élément*.

À l'intérieur de chaque sous-domaine, le champ est représenté par un nombre fini de valeurs évaluées en certains points choisis arbitrairement. Ces points sont appelés *nœuds*. Le champ local est une interpolation des valeurs aux nœuds. Le sous-domaine muni de son interpolation est appelé *élément*.

Chercher une solution par éléments finis consiste donc à déterminer quel champ local on attribue à chaque sous-domaine pour que le champ global obtenu par juxtaposition de ces champs locaux satisfasse au mieux les équations et les conditions aux limites.

La résolution d'un problème par la méthode des éléments finis se décompose selon les étapes suivantes :

❶ On divise le domaine Ω en sous-domaines : C'est le *maillage*. Les sous-domaines sont appelés *mailles*. En général, les mailles sont des primitives géométriques relativement simples, comme des tétraèdres ou des hexaèdres.

❷ Sur chaque sous-domaine, on définit un ensemble de nœuds. Ces nœuds correspondent aux points de contrôle sur lesquels le problème sera évalué. Ce sont en général des points remarquables de la primitive géométrique choisie. Dans les cas les plus simples ce sont ses sommets, mais si on veut augmenter le nombre de degrés de liberté, on peut aussi prendre les milieux des arêtes, les centres des faces, le centre de gravité, etc ...

❸ On choisit les polynômes (ou autres fonctions) qui définissent le champ local en fonction des valeurs aux nœuds. Ces polynômes sont aussi appelés *fonctions de forme* du sous-domaine. Une maille associée à ses nœuds et à ses fonctions de forme constitue un élément.

❹ On ramène ensuite le problème à un problème discret : c'est la discrétisation. En effet, toute solution approchée est complètement déterminée par les valeurs aux nœuds des éléments. Il suffit donc de trouver les valeurs à attribuer aux nœuds pour décrire une solution approchée.

❺ On résout le problème discret : c'est la *résolution*. Nous verrons les méthodes de résolution que nous utilisons dans le chapitre 2.

❻ On peut alors construire la solution approchée à partir des valeurs trouvées aux nœuds et en déduire d'autres grandeurs : c'est le *post-traitement*.

Nous allons maintenant détailler un peu la définition d'un élément fini de type P_1, qui est l'élément que nous utiliserons dans les deux chapitres suivants.

Les éléments finis de type P_1

La maille de ce type d'élément fini est le tétraèdre. C'est la primitive géométrique volumique la plus simple, puisqu'elle n'est composée que de quatre points. Pourtant le tétraèdre est très bien adapté à la représentation de formes géométriques complexes. De plus, la surface d'un maillage tétraédrique est une triangulation, qui permet de représenter de manière très fidèle une grande variété de surfaces.

L'élément P_1 sera donc composé de cette maille tétraédrique, associée aux quatre nœuds situés sur les sommets $\{\mathbf{P}_j \ , \ j = 0..3\}$ du tétraèdre et aux fonctions de formes linéaires $\{\Lambda_j \ , \ j = 0..3\}$ qui permettent d'exprimer en n'importe quel point $\mathbf{X}$ de l'élément toute fonction $\mathbf{h}$ à partir de ses valeurs aux sommets du tétraèdre :

$$\boxed{\mathbf{h}(\mathbf{X}) = \sum_{j=0}^{3} \Lambda_j(\mathbf{X})\ \mathbf{h}(\mathbf{P}_j)} \tag{1.30}$$

où $\Lambda_j(\mathbf{X}) = \boldsymbol{\alpha}_j \ . \ \mathbf{X} + \beta_j$.

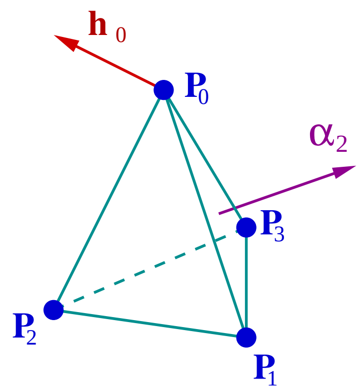


FIG. 1.3 – *Élément fini de type P_1*

Ces fonctions Λ_j correspondent aux coordonnées barycentriques du point $\mathbf{X}$ dans le tétraèdre. Pour trouver leurs expressions, on remplace dans l'équation 1.30 la fonction $\mathbf{h}$ par l'identité. De plus, on impose que la somme des coordonnées barycentriques soit égale à 1. Ceci nous donne un système de 4 équations à 4 inconnues, que l'on va écrire sous forme matricielle (les positions des sommets $\mathbf{P}_j$ ont pour composantes : $\mathbf{P}_j = [p_j \quad q_j \quad r_j]^t$) :

$$\left\{\begin{aligned} \mathbf{X} &= \sum_{j=0}^{3} \Lambda_j(\mathbf{X})\ \mathbf{P}_j \\ 1 &= \sum_{j=0}^{3} \Lambda_j(\mathbf{X}) \end{aligned}\right. \iff \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} p_0 & p_1 & p_2 & p_3 \\ q_0 & q_1 & q_2 & q_3 \\ r_0 & r_1 & r_2 & r_3 \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} \Lambda_0 \\ \Lambda_1 \\ \Lambda_2 \\ \Lambda_3 \end{bmatrix} \iff \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix} = \begin{bmatrix} P \end{bmatrix} \begin{bmatrix} \Lambda \end{bmatrix} \tag{1.31}$$

Les quatre sommets n'étant pas coplanaires (le tétraèdre ne doit pas être dégénéré pour que l'élément soit correct), la matrice P est inversible. Les fonctions Λ_j s'écrivent donc sous la forme :

$$\begin{bmatrix} \Lambda \end{bmatrix} = \begin{bmatrix} P^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix} = \frac{1}{det(P)}\ com(P)^t \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix} \iff \begin{bmatrix} \Lambda_0 \\ \Lambda_1 \\ \Lambda_2 \\ \Lambda_3 \end{bmatrix} = \begin{bmatrix} \boldsymbol{\alpha}_0 & \beta_0 \\ \boldsymbol{\alpha}_1 & \beta_1 \\ \boldsymbol{\alpha}_2 & \beta_2 \\ \boldsymbol{\alpha}_3 & \beta_3 \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix} \tag{1.32}$$

où $com(P)$ est la co-matrice de la matrice P. La définition de cette co-matrice permet d'exprimer les composantes des vecteur $\boldsymbol{\alpha}_j$ et les scalaires β_j de la manière suivante :

$$(\boldsymbol{\alpha}_j)_i = (-1)^{(j+i)}\ \frac{det[P']_{ji}}{det[P]} \quad \text{et} \quad \beta_j = (-1)^j\ \frac{det[P']_{j3}}{det[P]} \tag{1.33}$$

Les sous matrices $[P']_{ji}$ sont obtenues en retirant la $(i+1)^{\text{ème}}$ ligne et la $(j+1)^{\text{ème}}$ colonne de la matrice $[P]$. On remarque que le déterminant de la matrice $[P]$ est directement relié au volume V_T du tétraèdre. On peut alors montrer que le vecteur $\boldsymbol{\alpha}_j$, qui se nomme aussi vecteur de forme, peut s'exprimer de manière plus "intuitive" :

$$\boldsymbol{\alpha}_j = \frac{(-1)^j}{6\ V_T}\big(\mathbf{P}_{j+1} \wedge \mathbf{P}_{j+2} + \mathbf{P}_{j+2} \wedge \mathbf{P}_{j+3} + \mathbf{P}_{j+3} \wedge \mathbf{P}_{j+1}\big) = \frac{\mathbf{M}_j}{6\ V_T} \tag{1.34}$$

où $\mathbf{M}_j$ est un vecteur dirigé suivant la normale à la face opposée au sommet $\mathbf{P}_j$, et dont la norme est le double de l'aire de cette face.

1.4 Rappels sur les calculs vectoriels et matriciels

La mécanique des milieux continus et la méthode des éléments finis, que nous venons de présenter, et qui sont à la base des travaux décrits dans ce manuscrit, utilisent de manière intensive le calcul vectoriel et matriciel. Ce paragraphe en définit donc les notations et les opérateurs.

Ce sont, pour la plupart, des opérateurs classiques, mais dont les conventions de notation changent en fonction des auteurs et des domaines d'application. C'est pourquoi nous introduisons ici explicitement les notations que nous utiliserons. Le lecteur pourra alors s'y référer au cours de la lecture du manuscrit.

Nous nous plaçons dans un espace à 3 dimensions et nous posons les conventions de notations suivantes :

- Les scalaires et fonctions scalaires utilisent des caractères normaux minuscules. Par exemple $f(\mathbf{X}) = f(x,y,z)$ représente une fonction scalaire de $\mathbb{R}^3$ à valeurs dans $\mathbb{R}$.
- Les points, les vecteurs, les champs de vecteurs et les fonctions retournant des vecteurs sont représentés par des lettres minuscules ou majuscules en caractères gras. Certains de ces vecteurs reviendront assez souvent. C'est le cas par exemple de $\mathbf{X} = \mathbf{X}(x,y,z) = \begin{bmatrix} x & y & z \end{bmatrix}^t$ qui représente un point quelconque de l'espace $\mathbb{R}^3$, et de

$$\mathbf{\Phi} = \mathbf{\Phi}(x,y,z) = \begin{bmatrix} \phi(x,y,z) \\ \psi(x,y,z) \\ \zeta(x,y,z) \end{bmatrix} \qquad \text{et} \qquad \mathbf{U} = \mathbf{U}(x,y,z) = \begin{bmatrix} u(x,y,z) \\ v(x,y,z) \\ w(x,y,z) \end{bmatrix}, \tag{1.35}$$

  qui représentent respectivement un champ de déformations et un champ de vecteurs de $\mathbb{R}^3$. Néanmoins, certains vecteurs porteront une flèche afin d'attirer l'attention sur leur nature particulière. Ce sera le cas, par exemple, pour les vecteurs normaux ou tangents à une surface ($\overrightarrow{\mathbf{N}}$, $\overrightarrow{\mathbf{n}}$, ou $\overrightarrow{\mathbf{t}}$), ou pour les vecteurs joignant deux points ($\overrightarrow{\mathbf{P}_0\mathbf{P}_1}$).
- Les matrices sont identifiées par une lettre majuscule ou par une notation du type $[M_{ij}]$ où M_{ij} est le $(j+1)^{\text{ème}}$ élément de la $(i+1)^{\text{ème}}$ ligne (les numérotations des lignes, des colonnes et de tous les indices en général débutent à 0). Par exemple une matrice carrée (3x3) sera notée :

$$M = [M_{ij}] = \begin{bmatrix} M_{00} & M_{01} & M_{02} \\ M_{10} & M_{11} & M_{12} \\ M_{20} & M_{21} & M_{22} \end{bmatrix} \tag{1.36}$$

Avec ces notations, nous définissons les opérateurs classiques sur les vecteurs et les matrices. Pour plus de détails le lecteur pourra se rapporter au chapitre 4 de la thèse de Morten Bro-Nielsen (Bro-Nielsen, 1996), aux ouvrages (Bronshtein et Semendyayev, 1985; Rabbitt, 1997), au cours de Jean Garrigues à l'ESM2[1] (Garrigues, 1999c) ou à la très pratique encyclopédie des mathématiques en ligne *MathWorld* (Weisstein, 2000).

1. École Supérieure de Mécanique de Marseille : http://esm2.imt-mrs.fr/

Soit deux vecteurs $\mathbf{U}_1$ et $\mathbf{U}_2$ et deux matrices M et N. Nous considérons les opérateurs suivants :

- **Le produit scalaire de deux vecteurs :**

$$\mathbf{U}_1 \, . \, \mathbf{U}_2 = {\mathbf{U}_1}^t \, \mathbf{U}_2 = u_1 u_2 + v_1 v_2 + w_1 w_2 \tag{1.37}$$

- **Le produit vectoriel de deux vecteurs :**
À partir de deux vecteurs non colinéaires, on construit un troisième vecteur perpendiculaire aux deux premiers :

$$\mathbf{U}_1 \wedge \mathbf{U}_2 = \begin{bmatrix} v_1 w_2 - w_1 v_2 \\ w_1 u_2 - u_1 w_2 \\ u_1 v_2 - v_1 u_2 \end{bmatrix} \tag{1.38}$$

- **La trace d'une matrice :**
Notée $tr\ M$, la trace d'une matrice est la somme de ses éléments diagonaux :

$$tr\ M = \sum_i M_{ii} \tag{1.39}$$

- **Le produit tensoriel de deux vecteurs :**
À partir de deux vecteurs, on construit la matrice suivante :

$$\mathbf{U}_1 \otimes \mathbf{U}_2 = \big(\mathbf{U}_2 \otimes \mathbf{U}_1\big)^t = \mathbf{U}_1 \ {\mathbf{U}_2}^t = \begin{bmatrix} u_1 u_2 & u_1 v_2 & u_1 w_2 \\ v_1 u_2 & v_1 v_2 & v_1 w_2 \\ w_1 u_2 & w_1 v_2 & w_1 w_2 \end{bmatrix}. \tag{1.40}$$

Si $\mathbf{U}_1$ et $\mathbf{U}_2$ sont des vecteurs unitaires, la matrice ainsi obtenue peut être vue comme une application qui est la composition de la projection orthogonale sur le vecteur $\mathbf{U}_2$ et de la rotation qui donne à $\mathbf{U}_2$ la direction de $\mathbf{U}_1$. Cette application transforme donc le vecteur $\mathbf{X}$ en un vecteur colinéaire à $\mathbf{U}_1$ et de norme $\mathbf{X} \, . \, \mathbf{U}_2$.
On peut aussi remarquer que la trace du produit tensoriel de deux vecteurs est égale au produit scalaire de ces deux vecteurs :

$$tr\big(\mathbf{U}_1 \otimes \mathbf{U}_2\big) = u_1 u_2 + v_1 v_2 + w_1 w_2 = \mathbf{U}_1 \, . \, \mathbf{U}_2 \tag{1.41}$$

- **Le déterminant d'une matrice :**
On le notera $det\ M$ ou $|M|$. Pour une matrice 3x3, il vaut :

$$\begin{aligned} det\ M = |M| = & M_{00} M_{11} M_{22} + M_{01} M_{12} M_{20} + M_{02} M_{10} M_{21} \\ & - M_{02} M_{11} M_{20} - M_{00} M_{12} M_{21} - M_{01} M_{10} M_{22} \end{aligned} \tag{1.42}$$

- **La transposée d'une matrice :**
La matrice transposée de M, notée M^t, est la matrice obtenue en inversant les lignes et les colonnes de M :

$$M^t = [M_{ij}]^t = [M_{ji}] = \begin{bmatrix} M_{00} & M_{10} & M_{20} \\ M_{01} & M_{11} & M_{21} \\ M_{02} & M_{12} & M_{22} \end{bmatrix} \tag{1.43}$$

Une matrice égale à sa transposée est symétrique.

- **Le produit scalaire de deux matrices :**
  On définit l'opérateur " : ", qui appliqué à deux matrices M et N, donne :

$$M : N = tr(M^t N) = tr(M N^t) = \sum_{ij} M_{ij} N_{ij} \tag{1.44}$$

  On vérifie que cet opérateur est bien bilinéaire, symétrique, défini, positif, et qu'il définit donc un produit scalaire de matrices. Ce produit scalaire engendre donc une norme de matrice :

$$\|M\|^2 = M : M = tr(M^t M) = tr(M M^t) = \sum_{ij} M_{ij}^2 \tag{1.45}$$

1.5 Rappels sur les opérateurs différentiels

La mécanique des milieux continus amène à manipuler des champs, qu'ils soient scalaires (champs de potentiels) ou vectoriels (champs de déplacements et de déformations). Nous introduisons dans ce paragraphe des opérateurs différentiels que nous allons être amenés à appliquer sur ces champs.

- Tout d'abord, de nombreux calculs feront intervenir des dérivées partielles. Nous indiquerons alors en indice le nom de la variable par rapport à laquelle l'objet a été dérivé. Ainsi

$$\blacksquare_x = \frac{\partial \blacksquare}{\partial x} \tag{1.46}$$

  est la dérivée partielle de $\blacksquare$ par rapport à x.
- **La dérivation par rapport à une matrice :**
  On peut aussi dériver une fonction scalaire par rapport à une matrice. On obtient alors une matrice dont chaque élément est la dérivée partielle de la fonction par rapport à l'élément correspondant de la matrice :

$$\frac{\partial f}{\partial M} = \left[\frac{\partial f}{\partial M_{ij}} \right] \tag{1.47}$$

- **Le vecteur ∇ :**
  Tous les opérateurs suivants, basés sur des dérivations partielles, reposent sur le vecteur ∇, qui est défini par :

$$\nabla = \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} \end{bmatrix} \tag{1.48}$$

- **L'opérateur gradient :**
  Appliqué à une fonction **scalaire** f, il engendre le **vecteur gradient** de la fonction :

$$\nabla f = \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} \tag{1.49}$$

Appliqué à un champ de **vecteurs U**, il engendre le **tenseur gradient** du champ de vecteurs :

$$\nabla \mathbf{U} = \left(\nabla\ \mathbf{U}^t\right)^t = \begin{bmatrix} u_x & u_y & u_z \\ v_x & v_y & v_z \\ w_x & w_y & w_z \end{bmatrix} \tag{1.50}$$

- **L'opérateur divergence :**
  Il s'applique à un vecteur pour donner un scalaire :

$$div\ \mathbf{U} = \nabla\ .\ \mathbf{U} = \nabla^t \mathbf{U} = u_x + v_y + w_z = tr(\nabla \mathbf{U}) \tag{1.51}$$

On le généralise à la divergence d'une matrice, qui donne un vecteur. Les composantes du vecteur sont alors les divergences scalaires des vecteurs formés par les colonnes de la matrice :

$$div\ M = \nabla\ .\ M = (\nabla^t M)^t = \begin{bmatrix} (M_{00})_x + (M_{10})_y + (M_{20})_z \\ (M_{01})_x + (M_{11})_y + (M_{21})_z \\ (M_{02})_x + (M_{12})_y + (M_{22})_z \end{bmatrix} \tag{1.52}$$

! **Remarque :** Avec les notations proposées pour le gradient d'un vecteur et la divergence d'une matrice, on retrouve la propriété de commutation de ces opérateurs :

$$\begin{aligned} \nabla(div\ \mathbf{U}) &= \nabla(\nabla.\mathbf{U}) = \nabla\nabla^t\mathbf{U} \\ &= \begin{bmatrix} u_{xx} + v_{xy} + w_{xz} \\ u_{xy} + v_{yy} + w_{yz} \\ u_{xz} + v_{yz} + w_{zz} \end{bmatrix} = \left(\begin{bmatrix} \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \end{bmatrix} \begin{bmatrix} u_x & u_y & u_z \\ v_x & v_y & v_z \\ w_x & w_y & w_z \end{bmatrix} \right)^t \\ &= \left(\nabla^t(\nabla\mathbf{U})\right)^t = div(\nabla\mathbf{U}) \end{aligned} \tag{1.53}$$

- **L'opérateur rotationnel :**
  Le rotationnel d'un vecteur est le produit vectoriel du vecteur ∇ par ce vecteur :

$$rot\ \mathbf{U} = \nabla \wedge \mathbf{U} = \begin{bmatrix} w_y - v_z \\ u_z - w_x \\ v_x - u_y \end{bmatrix} \tag{1.54}$$

- **Le laplacien :**
  Lorsque l'on prend la divergence du gradient, on obtient le laplacien. Dans le cas d'une fonction scalaire f on obtient :

$$\Delta f = div(\nabla f) = \nabla^t \nabla f = f_{xx} + f_{yy} + f_{zz} \tag{1.55}$$

Avec les notations précédentes, on retrouve la généralisation au laplacien vectoriel, qui est le vecteur formé par le laplacien de chaque composante du vecteur :

$$\Delta \mathbf{U} = div(\nabla \mathbf{U}^t) = \left| \begin{array}{c} div(\nabla \mathbf{U}) \\ = \\ \nabla(div\ \mathbf{U}) \end{array} \right| - rot(rot\ \mathbf{U})$$
$$= \begin{bmatrix} \Delta u \\ \Delta v \\ \Delta w \end{bmatrix} = \begin{bmatrix} u_{xx} + u_{yy} + u_{zz} \\ v_{xx} + v_{yy} + v_{zz} \\ w_{xx} + w_{yy} + w_{zz} \end{bmatrix} \tag{1.56}$$

1.6 Conclusion

Ce premier chapitre nous a permis de poser les bases mathématiques et physiques qui seront utilisées dans les chapitres suivants. Nous y avons introduit les principes de la mécanique des milieux continus, puis de la théorie de l'élasticité, ce qui nous a mené jusqu'à la présentation du modèle d'élasticité de St Venant-Kirchhoff. Nous avons ensuite présenté la méthode des éléments finis que nous utiliserons pour résoudre les problèmes élastiques posés par la déformation des organes virtuels. Enfin, nous avons consacré la dernière partie de ce chapitre à la description des notations et des propriétés des calculs vectoriels, matriciels et différentiels.

Toutes ces notions seront principalement utilisées dans les deux chapitres suivants, qui décriront les modèles déformables que nous proposons : dans le chapitre 2 nous nous intéresserons à la mise en œuvre et à l'enrichissement du modèle élastique linéaire, alors que dans le chapitre 3 nous nous concentrerons sur l'étude de l'élasticité non-linéaire (grands déplacements) de St Venant-Kirchhoff.

Chapitre 2

L'élasticité linéaire

Ce chapitre est consacré à la mise en œuvre de plusieurs modèles déformables basés sur la théorie de l'élasticité linéaire et la méthode des éléments finis. Nous commençons par présenter deux modèles déformables qui diffèrent par la méthode de résolution qu'ils utilisent, à savoir une méthode implicite quasi-statique couplée à l'utilisation de pré-calculs, et une méthode explicite aboutissant à un modèle dynamique. Nous proposons ensuite d'enrichir le réalisme biomécanique de ces modèles en y introduisant la notion d'anisotropie, afin de se rapprocher du comportement des tissus biologiques.

2.1 Introduction

Ce chapitre est exclusivement consacré à la mise en œuvre de ce qui demeure le centre névralgique de tout simulateur de chirurgie : **la modélisation physique** des corps déformables. En effet, ce que nous appelons un modèle déformable a la lourde tâche de représenter, de la manière la plus réaliste possible, l'organe sur lequel l'utilisateur va venir simuler des gestes chirurgicaux. Il doit pour cela avoir l'apparence de l'organe qu'il modélise (il doit en respecter la géométrie, la couleur et la texture), mais aussi et surtout avoir le même comportement biomécanique, c'est-à-dire se déformer comme le ferait le véritable organe. Il doit même pouvoir se découper ou se déchirer le cas échéant. Mais le point le plus délicat est que ce réalisme ne doit pas nuire à l'autre qualité première d'un simulateur de chirurgie : **l'interactivité**. En effet, pour que l'utilisateur "accepte" de croire à la réalité de ce qu'il voit, il faut avant tout que les mouvements et les déformations soient fluides et semblent continus. De plus, il ne doit percevoir aucun décalage entre les gestes qu'il effectue et ce qu'il voit sur l'écran. Pour cela, Il faut au moins que le simulateur respecte la contrainte du temps-réel visuel, qui est de 25 images par seconde (25 Hz), et que la latence soit la plus faible possible. Toute la réflexion autour de l'élaboration du modèle déformable sera donc centrée sur le choix du meilleur compromis possible entre le réalisme (biomécanique et visuel) et l'efficacité.
La construction d'un modèle déformable se fait en plusieurs étapes :

❶ Il faut, tout d'abord, un modèle géométrique qui représente l'organe auquel on s'intéresse. Nous travaillons avec des maillages tétraédriques, particulièrement bien adaptés pour modéliser des objets à la géométrie complexe. Nous détaillerons la construction de ces maillages au chapitre 5.

❷ Il faut ensuite choisir le comportement biomécanique que l'on désire donner à l'organe en question. Afin de garantir le réalisme physique de ce comportement, nous utilisons des lois de déformation issues de la mécanique des milieux continus. Dès lors que le temps de calcul est un facteur important, l'élasticité linéaire semble le modèle le plus approprié. En effet, sa linéarité lui confère une bonne efficacité et permet de lui appliquer plusieurs méthodes d'accélération et d'optimisation, dont certaines seront présentées dans ce chapitre. De plus, la relative simplicité des équations qu'il engendre permet une interprétation "intuitive" des différents comportements, ce qui permet d'y apporter de nombreuses améliorations.

❸ L'étape suivante consiste à discrétiser le problème élastique obtenu afin de permettre le calcul d'une solution numérique. Lors de cette discrétisation, il faut prévoir la possibilité d'appliquer sur le modèle un ensemble de contraintes aux bords. C'est par l'intermédiaire de ces contraintes que l'utilisateur pourra interagir avec l'organe virtuel.

❹ Enfin, il faut résoudre le problème, c'est-à-dire calculer la déformation subie par l'organe et éventuellement en déduire d'autres informations utiles, comme par exemple les contraintes dans les zones de contact entre l'organe et le monde extérieur.

Dans ce chapitre, nous présentons dans un premier temps deux modèles déformables fondés sur l'élasticité linéaire et la méthode des éléments finis, qui sont inspirés des travaux de S. Cotin (Cotin, 1997). Ces deux modèles n'utilisent pas la même méthode de résolution. Le premier met en œuvre un schéma de résolution implicite quasi-statique permettant de construire des pré-calculs qui seront utilisés lors de la simulation : c'est le modèle pré-calculé. Le second utilise une méthode explicite et un comportement dynamique : c'est le modèle masse-tenseur. Pour chacun d'entre eux, nous détaillons la méthode de résolution, ainsi que les particularités qui font leurs forces ou leurs faiblesses. Nous proposons ensuite d'enrichir le comportement biomécanique de ces deux modèles en y introduisant la notion de déformation anisotrope. Pour cela, nous généralisons le modèle élastique linéaire aux matériaux transversalement isotropes, c'est-à-dire possédant une direction privilégiée de déformation. Nous appliquons ce nouveau modèle pour la simulation de structures fortement anisotropes (comme les tendons) ou de tissus traversés par des réseaux de fibres. Il nous permet aussi de simuler la présence de gros vaisseaux sanguins à l'intérieur du foie. Enfin, toujours dans le but d'améliorer le réalisme biomécanique des déformations simulées, nous avons ajouté à notre modèle masse-tenseur une surface plus rigide (anisotropie de surface), afin de prendre en compte la présence d'une peau élastique à la surface de certains organes, comme la capsule de Glisson qui entoure le foie.

2.2 Énergie potentielle élastique : loi de *Hooke*

L'élasticité linéaire est un cas particulier du modèle de St Venant-Kirchhoff. En effet, en remplaçant le tenseur E par sa version linéarisée (équation 1.9) dans l'équation de l'énergie élastique 1.26, on obtient la *loi de Hooke* :

$$W^l = \frac{\lambda}{2}\,(trE_l)^2 + \mu\; trE_l{}^2. \tag{2.1}$$

En utilisant les composantes de déformation introduites dans l'équation 1.9 et les définitions des opérateurs *gradient*, *divergence* et *rotationnel*, on peut exprimer l'énergie W^l en fonction du champ de déplacements par :

$$\begin{aligned}
W^l =& \frac{\lambda}{2}(\epsilon^l_x + \epsilon^l_y + \epsilon^l_z)^2 + \mu({\epsilon^l_x}^2 + {\epsilon^l_y}^2 + {\epsilon^l_z}^2) + \frac{\mu}{2}({\gamma^l_{xy}}^2 + {\gamma^l_{xz}}^2 + {\gamma^l_{yz}}^2)\\
=& \frac{\lambda}{2}(u_x + v_y + w_z)^2 + \mu(u_x^2 + v_y^2 + w_z^2)\\
&+\frac{\mu}{2}(u_y^2 + u_z^2 + v_x^2 + v_z^2 + w_x^2 + w_y^2 + 2u_yv_x + 2u_zw_x + 2v_zw_y)\\
=& \frac{\lambda}{2}(u_x + v_y + w_z)^2 + \mu(u_x^2 + u_y^2 + u_z^2 + v_x^2 + v_y^2 + v_z^2 + w_x^2 + w_y^2 + w_z^2)\\
&-\frac{\mu}{2}(u_y^2 + u_z^2 + v_x^2 + v_z^2 + w_x^2 + w_y^2 - 2u_yv_x - 2u_zw_x - 2v_zw_y)\\
W^l =& \frac{\lambda}{2}(div\ \mathbf{U})^2 + \mu\|\nabla\mathbf{U}\|^2 - \frac{\mu}{2}\|rot\ \mathbf{U}\|^2
\end{aligned} \tag{2.2}$$

Si l'on applique les principes du calcul variationnel, et les propriétés des opérateurs présents dans l'équation 2.2, on obtient l'équation aux dérivées partielles (EDP) gouvernant les déformations linéaires élastiques dans le cas statique (annexe A.1) :

$$\mu\,\Delta\mathbf{U} + (\lambda+\mu)\,\nabla\big(div\ \mathbf{U}\big) + \mathbf{f} = \mathbf{0}, \tag{2.3}$$

où $\mathbf{f}$ est le champ de forces volumiques externes appliquées sur le domaine. On vérifie bien que cette équation est linéaire par rapport aux dérivées partielles du champ de déplacements.

Les solutions analytiques des problèmes d'élasticité linéaire n'étant envisageables que dans un certain nombre de cas très particuliers, il nous faut discrétiser ce problème afin d'en calculer une solution numérique approchée. Pour cela nous utilisons la méthode des éléments finis.

2.3 L'élasticité linéaire en éléments finis

Nous avons vu dans le paragraphe 1.3 que la méthode des éléments finis consiste à subdiviser le domaine sur lequel on travaille pour ensuite approximer, sur chaque sous domaine, le problème que l'on se pose en l'exprimant comme une interpolation de valeurs calculées en certains points. Pour les éléments finis de type P_1, chaque sous domaine est un tétraèdre dans lequel le problème est approximé par l'interpolation linéaire de ses valeurs sur les quatre sommets.

Nous considérons donc un maillage tétraédrique de l'organe qui nous intéresse, sur lequel nous allons discrétiser l'énergie élastique de déformation exprimée par l'équation 2.1. La première étape consiste à exprimer la valeur du champ de déplacements $\mathbf{U}$ à l'intérieur de chaque tétraèdre en fonction des déplacements de ses sommets. Pour cela, on intègre $\mathbf{U}$ dans l'équation 1.30 :

$$\boxed{\mathbf{U}(\mathbf{X}) = \sum_{j=0}^{3} \Lambda_j(\mathbf{X})\ \mathbf{U}_j} \tag{2.4}$$

où $\begin{cases} \Lambda_j(\mathbf{X}) = \boldsymbol{\alpha}_j\ .\ \mathbf{X} + \beta_j,\ j = 0..3, \\ \text{sont les fonctions d'interpolation linéaires, et} \\ \\ \mathbf{U}_j = \mathbf{U}(\mathbf{P}_j),\ j = 0..3, \\ \text{sont les déplacements des sommets.} \end{cases}$

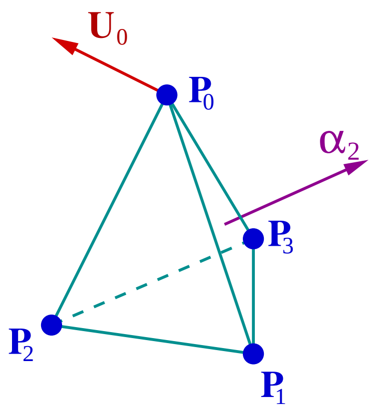


FIG. 2.1 – *Élément fini de type P_1*

Nous pouvons ainsi exprimer le gradient du champ de déplacement :

$$\nabla\mathbf{U} = \sum_{j=0}^{3} \mathbf{U}_j\ \nabla\left(\boldsymbol{\alpha}_j\ .\ \mathbf{X} + \beta_j\right) = \sum_{j=0}^{3} \mathbf{U}_j\ \boldsymbol{\alpha}_j^t$$

$$\Longrightarrow \boxed{\nabla\mathbf{U} = \sum_{j=0}^{3} \mathbf{U}_j \otimes \boldsymbol{\alpha}_j} \tag{2.5}$$

On remarque que ce gradient est constant à l'intérieur du tétraèdre, et qu'il sera donc représenté sur tout le maillage par une fonction constante par morceaux.

On va maintenant pouvoir exprimer le tenseur de déformation E_l, les deux termes $(tr\ E_l)^2$ et $tr\ {E_l}^2$, pour finalement obtenir l'expression de l'énergie élastique de déformation à l'intérieur d'un tétraèdre :

$$E_l = \frac{1}{2}\left(\nabla \mathbf{U}^t + \nabla \mathbf{U}\right) = \frac{1}{2}\sum_{j=0}^{3}\left(\mathbf{U}_j \otimes \boldsymbol{\alpha}_j + \boldsymbol{\alpha}_j \otimes \mathbf{U}_j\right) \tag{2.6}$$

$$\left(tr\ E_l\right)^2 = \left(\sum_{j=0}^{3} \mathbf{U}_j \ .\ \boldsymbol{\alpha}_j\right)^2 = \sum_{j,k=0}^{3} \mathbf{U}_j^t \left[\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k\right] \mathbf{U}_k \tag{2.7}$$

$$\begin{aligned} tr\ {E_l}^2 &= \frac{1}{4} tr\Bigg(\sum_{j,k=0}^{3} \left(\mathbf{U}_j \otimes \boldsymbol{\alpha}_k\right)\left(\mathbf{U}_k \ .\ \boldsymbol{\alpha}_j\right) + \left(\boldsymbol{\alpha}_j \otimes \mathbf{U}_k\right)\left(\mathbf{U}_j \ .\ \boldsymbol{\alpha}_k\right) \\ &\qquad + \left(\mathbf{U}_j \otimes \mathbf{U}_k\right)\left(\boldsymbol{\alpha}_j \ .\ \boldsymbol{\alpha}_k\right) + \left(\mathbf{U}_j \ .\ \mathbf{U}_k\right)\left(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k\right)\Bigg) \\ &= \frac{1}{2}\sum_{j,k=0}^{3} \mathbf{U}_j^t \left[\left(\boldsymbol{\alpha}_k \otimes \boldsymbol{\alpha}_j\right) + \left(\boldsymbol{\alpha}_j \ .\ \boldsymbol{\alpha}_k\right) Id\right] \mathbf{U}_k \end{aligned} \tag{2.8}$$

$$\boxed{W^l(\mathcal{T}) = \frac{1}{2}\sum_{j,k=0}^{3} \mathbf{U}_j^t \left[\mathcal{B}_{\mathcal{T}}^{jk}\right] \mathbf{U}_k \quad \text{avec} \quad \mathcal{B}_{\mathcal{T}}^{jk} = \lambda\left(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k\right) + \mu\Big[\left(\boldsymbol{\alpha}_k \otimes \boldsymbol{\alpha}_j\right) + \left(\boldsymbol{\alpha}_j \ .\ \boldsymbol{\alpha}_k\right) Id\Big]} \tag{2.9}$$

Il est intéressant de remarquer que :

$$\begin{aligned} \boxed{\left[\mathcal{B}_{\mathcal{T}}^{kj}\right]} &= \lambda\left(\boldsymbol{\alpha}_k \otimes \boldsymbol{\alpha}_j\right) + \mu\Big[\left(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k\right) + \left(\boldsymbol{\alpha}_k \ .\ \boldsymbol{\alpha}_j\right) Id\Big] \\ &= \lambda\left(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k\right)^t + \mu\Big[\left(\boldsymbol{\alpha}_k \otimes \boldsymbol{\alpha}_j\right)^t + \left(\boldsymbol{\alpha}_j \ .\ \boldsymbol{\alpha}_k\right) Id\Big] \\ & \boxed{= \left[\mathcal{B}_{\mathcal{T}}^{jk}\right]^t} \end{aligned} \tag{2.10}$$

L'ensemble des matrices (3x3) $\left\{\mathcal{B}_{\mathcal{T}}^{jk}\ ;\ j,k = 0..3\right\}$ constitue l'ensemble des tenseurs locaux de rigidité du tétraèdre $\mathcal{T}$. Ils représentent les caractéristiques locales du matériau élastique. Si $j = k$, on parle du tenseur local de rigidité du sommet $\mathbf{P}_j$, sinon du tenseur local de rigidité de l'arête $(\mathbf{P}_j, \mathbf{P}_k)$. On montre facilement que $[\mathcal{B}_{\mathcal{T}}^{kj}] = [\mathcal{B}_{\mathcal{T}}^{jk}]^t$. De ce fait, seulement 10 tenseurs sont nécessaires pour décrire le comportement du matériau à l'intérieur de ce tétraèdre.

En dérivant cette énergie de déformation par rapport à la position d'un des sommets $\mathbf{P}_p$, on obtient la force interne $\mathbf{F}_p^l(\mathcal{T})$ exercée sur ce sommet par l'élément de matière élastique que représente le tétraèdre $\mathcal{T}$. On se rend alors compte que cette force peut se décomposer en deux parties. La première partie est la force créée par le déplacement du sommet lui-même. Elle a tendance à rappeler le sommet vers sa position de repos. La deuxième force est engendrée par le déplacement des autres sommets du tétraèdre. Elle traduit l'influence que les sommets exercent sur leurs voisins. C'est donc elle qui va permettre la propagation de la déformation à l'intérieur du matériau. La donnée des forces élastiques exercées en chaque sommet permet de caractériser le comportement élastique linéaire de ce tétraèdre :

$$\mathbf{F}_p^l(\mathcal{T}) = \sum_{j=0}^{3} [\mathcal{B}_{\mathcal{T}}^{pj}]\,\mathbf{U}_j$$

$$\Longleftrightarrow \boxed{\mathbf{F}_p^l(\mathcal{T}) = [\mathcal{B}_{\mathcal{T}}^{pp}]\,\mathbf{U}_p + \sum_{\substack{j=0\\ j\neq p}}^{3} [\mathcal{B}_{\mathcal{T}}^{pj}]\,\mathbf{U}_j} \quad (2.11)$$

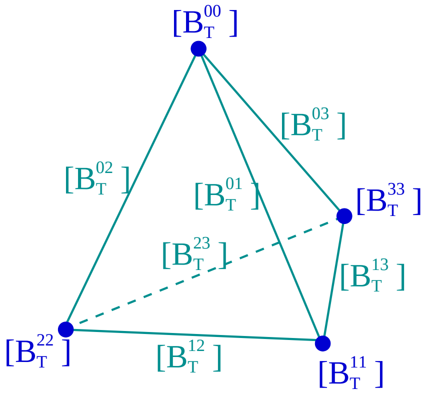


FIG. 2.2 – *Répartition des tenseurs de rigidité sur les sommets et sur les arêtes du tétraèdre*

Si on considère maintenant non plus un seul tétraèdre mais le maillage tétraédrique du domaine tout entier, on doit accumuler les contributions de chaque élément. La force interne globale $\mathbf{F}_p^l$ exercée sur un sommet se calcule alors comme la somme des forces internes calculées sur chaque tétraèdre adjacent à ce sommet. En posant cette somme, on s'aperçoit que l'on peut mettre en facteur le déplacement du sommet et celui de chacun de ses voisins pour finalement obtenir l'équation de la force interne sous la forme :

$$\begin{aligned}
\mathbf{F}_p^l &= \sum_{\mathcal{T}\,\in\,\nu(\mathbf{P}_p)} \mathbf{F}_p^l(\mathcal{T}) = \sum_{\mathcal{T}\,\in\,\nu(\mathbf{P}_p)} \left([\mathcal{B}_{\mathcal{T}}^{pp}]\,\mathbf{U}_p + \sum_{\substack{j=0\\ j\neq p}}^{3} [\mathcal{B}_{\mathcal{T}}^{pj}]\,\mathbf{U}_j \right) \\
&= \underbrace{\left(\sum_{\mathcal{T}\,\in\,\nu(\mathbf{P}_p)} [\mathcal{B}_{\mathcal{T}}^{pp}] \right)}_{[\mathcal{B}^{pp}]} \mathbf{U}_p + \sum_{\mathcal{A}\,\in\,\nu(\mathbf{P}_p)} \underbrace{\left(\sum_{\mathcal{T}\,\in\,\nu(\mathbf{P}_p\,,\,\mathbf{P}_j)} [\mathcal{B}_{\mathcal{T}}^{pj}] \right)}_{[\mathcal{B}^{pj}]} \mathbf{U}_j
\end{aligned} \quad (2.12)$$

Avec les notations suivantes pour les sommes :

- $\mathcal{T} \in \nu(\mathbf{P}_p)$: *"somme sur tous les tétraèdres appartenant au voisinage du sommet* $\mathbf{P}_p$*"*,
- $\mathcal{A} \in \nu(\mathbf{P}_p)$: *"somme sur toutes les arêtes appartenant au voisinage du sommet* $\mathbf{P}_p$*"*, et
- $\mathcal{T} \in \nu(\mathbf{P}_p, \mathbf{P}_j)$: *"somme sur tous les tétraèdres appartenant au voisinage de l'arête* $(\mathbf{P}_p, \mathbf{P}_j)$*"*.

Lors de la construction de la structure de donnée liée au maillage, nous allons calculer, pour chaque tétraèdre, les tenseurs de rigidité locaux et accumuler leurs contributions sur les sommets et les arêtes. Sur la figure 2.3, le tenseur de rigidité du sommet identifié par un point ou de l'arête centrale sont la somme des contributions de tous les tétraèdres adjacents. On obtient ainsi des tenseurs de rigidité **globaux** pour chaque sommet ($\{\mathcal{B}^{pp}\}$) et pour chaque arête ($\{\mathcal{B}^{pj}, p \neq j\}$) du maillage.

La force interne globale exercée sur chaque nœud du maillage s'exprime donc en fonction du

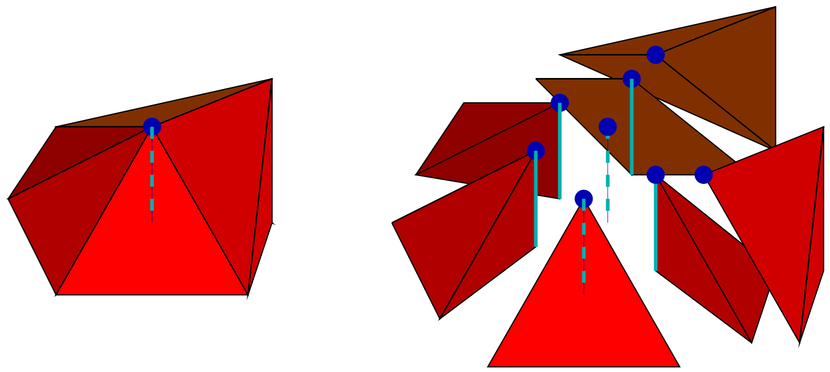


FIG. 2.3 – *Accumulation des tenseurs de rigidité sur les sommets et les arêtes du maillage*

déplacement de ce nœud et des déplacements de ses voisins par la formule suivante :

$$\boxed{\mathbf{F}^l_p = \left[\mathcal{B}^{pp}\right]\mathbf{U}_p + \sum_{\mathcal{A}\,\in\,\nu(\mathbf{P}_p)} \left[\mathcal{B}^{pj}\right]\mathbf{U}_j} \tag{2.13}$$

2.4 Les méthodes de résolution

Soit N_s le nombre de sommets du maillage. Nous disposons alors d'un ensemble de N_s équations du type 2.13 exprimant les forces élastiques internes exercées sur chacun des nœuds du maillage en fonction du champ de déplacements local. Chaque équation étant vectorielle, nous avons affaire à un système de $3 * N_s$ équations à $3 * N_s$ inconnues.

Avant de pouvoir résoudre ce problème, nous allons devoir modifier certaines équations afin de prendre en compte les conditions aux bords et les contraintes que nous voulons appliquer sur notre modèle :

- certains nœuds peuvent être fixés, ce qui correspond pour les organes du corps humain, aux zones en contact avec des structures voisines ou directement attachées sur les os.
- On peut aussi imposer un déplacement non nul, ce qui revient à pré-contraindre le modèle.
- Enfin, on peut appliquer des forces sur les sommets de la surface (forces de pression ou de contact) ou sur les sommets internes comme par exemple lorsque l'on veut modéliser l'action de la gravité.

On pourra alors résoudre le problème global et en déduire la déformation du modèle, c'est-à-dire calculer l'ensemble des déplacements des nœuds du maillage.

Dans cette partie, nous allons présenter les deux méthodes de résolution que nous utilisons. La première met en œuvre une résolution implicite du problème, accélérée par l'utilisation de pré-calculs, alors que la seconde utilise un schéma explicite dynamique. Ces deux méthodes de résolution donnent naissance à deux **modèles physiques déformables**, le modèle pré-calculé et le modèle masse-tenseur.

2.4.1 Résolution implicite : modèle pré-calculé

Il est classique de présenter le problème de l'élasticité linéaire discrétisée par la méthode des éléments finis sous la forme d'un unique système d'équations linéaires qui s'écrit sous la forme $[K]\mathbf{U} = \mathbf{F}$. L'ensemble $\mathbf{U}$ des déplacements des sommets du maillage sera alors obtenu en résolvant le système par une méthode itérative du type *gradient conjugué*. Malheureusement, le temps de calcul de ce type de résolution rend son utilisation impossible dans le cadre d'une application temps-réel. Par exemple, avec un maillage comprenant 1313 sommets, ce qui donne un système linéaire (3939 x 3939), il faut 140 itérations de gradient conjugué[1] pré-conditionné (factorisation de Cholesky ou LU incomplètes) pour atteindre une précision de 10^{-6} m, ce qui prend environ 9 secondes sur un PC Pentium II (450 MHz) sous linux.

Pour contourner ce problème de temps de calcul, nous utilisons une méthode dérivée de celle développée par Cotin et al. (Cotin, 1997; Cotin et al., 1999). Cette méthode est basée sur le fait que, grâce à la linéarité du modèle, la déformation obtenue en appliquant la somme de deux contraintes est bien la somme des déformations obtenues en appliquant chaque contrainte indépendamment (figure 2.4).

Donc, à partir du moment où l'on a calculé les déformations engendrées par l'application de certaines contraintes, on est capable de retrouver directement les déformations correspondant à l'application d'une combinaison linéaire de ces contraintes. L'idée consiste alors à pré-calculer une base de déformations élémentaires et à se servir de cette base pour exprimer n'importe quelle déformation du modèle.

2.4.1.1 Construction de la base de déformations

La première chose à faire est d'assembler la matrice de rigidité globale du système à partir des matrices de rigidité élémentaires des sommets et des arêtes (les $\mathcal{B}^{pp}$ et $\mathcal{B}^{pj}$ de l'équation 2.13). Cette matrice K est de dimension ($3N_s$ x $3N_s$) (où N_s est le nombre de sommets du maillage). Une fois cette matrice assemblée, le problème peut s'écrire sous la forme $[K]\ \mathbf{U} = \mathbf{F}$, ou encore :

$$\begin{bmatrix} \mathcal{B}^{0,0} & \cdots & \mathcal{B}^{0,p} & \cdots & \mathcal{B}^{0,N_s-1} \\ \vdots & \ddots & \vdots & & \vdots \\ \mathcal{B}^{p,0} & \cdots & \mathcal{B}^{p,p} & \cdots & \mathcal{B}^{p,N_s-1} \\ \vdots & & \vdots & \ddots & \vdots \\ \mathcal{B}^{N_s-1,0} & \cdots & \mathcal{B}^{N_s-1,p} & \cdots & \mathcal{B}^{N_s-1,N_s-1} \end{bmatrix} \begin{bmatrix} \mathbf{U}_0 \\ \vdots \\ \mathbf{U}_p \\ \vdots \\ \mathbf{U}_{N_s-1} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_0 \\ \vdots \\ \mathbf{F}_p \\ \vdots \\ \mathbf{F}_{N_s-1} \end{bmatrix} \tag{2.14}$$

Il faut remarquer que la matrice K est creuse puisque la matrice élémentaire $\mathcal{B}^{pj}$ n'est non nulle que si il existe une arête reliant les sommets $\mathbf{P}_p$ et $\mathbf{P}_j$. Or dans un maillage tétraédrique régulier, chaque sommet n'est relié qu' à une dizaine d'arêtes. Il faut donc tenir compte de cette propriété pour optimiser les temps de calcul, en utilisant des structures de données adaptées aux systèmes creux.

1. Nous utilisons la librairie d'algèbre linéaire *"Matrix Template Library"* : http://www.lsc.nd.edu/research/mtl/ et http://www.lsc.nd.edu/research/itl/

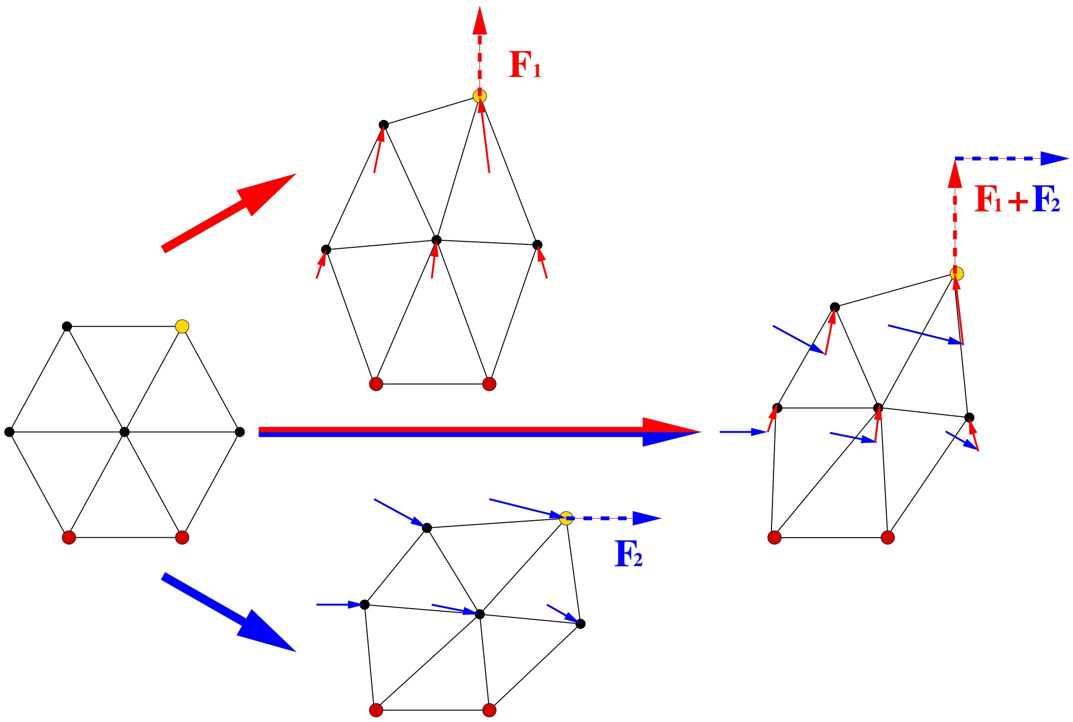


FIG. 2.4 – *À gauche se trouve le maillage au repos. Les deux sommets du bas sont fixés, et on applique les forces sur le sommet en haut à droite. Les déplacements obtenus en appliquant la somme des deux forces* $\mathbf{F}_1 + \mathbf{F}_2$ *(à droite) sont égaux à la somme des déplacements obtenus en appliquant indépendamment la force* $\mathbf{F}_1$ *(en haut) et la force* $\mathbf{F}_2$ *(en bas).*

C'est en partie pour cette raison que nous avons choisi d'utiliser le librairie de calcul algébrique "Matrix Template Library" (Lumsdaine et Siek, 1998; Lumsdaine et al., 1998).

Il faut ensuite définir des conditions aux limites, ce qui revient à modifier le système linéaire 2.14. Ces modifications sont plus ou moins compliquées suivant le cas. Le plus simple est d'imposer des forces $\mathbf{F}_p^*$ sur certains sommets. Il suffit pour cela de modifier le second membre du système :

$$\begin{bmatrix} \mathcal{B}^{0,0} & \cdots & \mathcal{B}^{0,p} & \cdots & \mathcal{B}^{0,N_s-1} \\ \vdots & \ddots & \vdots & & \vdots \\ \mathcal{B}^{p,0} & \cdots & \mathcal{B}^{p,p} & \cdots & \mathcal{B}^{i,N_s-1} \\ \vdots & & \vdots & \ddots & \vdots \\ \mathcal{B}^{N_s-1,0} & \cdots & \mathcal{B}^{N_s-1,p} & \cdots & \mathcal{B}^{N_s-1,N_s-1} \end{bmatrix} \begin{bmatrix} \mathbf{U}_0 \\ \vdots \\ \mathbf{U}_p \\ \vdots \\ \mathbf{U}_{N_s-1} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_0 \\ \vdots \\ \boxed{\mathbf{F}_p^*} \\ \vdots \\ \mathbf{F}_{N_s-1} \end{bmatrix} \tag{2.15}$$

Mais on peut aussi imposer les déplacements $\mathbf{U}_p^*$ de certains noeuds. Cette fois, il faut modifier la matrice de rigidité. En effet, imposer le déplacement d'un sommet revient à enlever sa contribution dans la déformation, en la redistribuant sous forme de forces dans le second membre :

$$\begin{bmatrix} \mathcal{B}^{0,0} & \cdots & \begin{smallmatrix}0&0&0\\0&0&0\\0&0&0\end{smallmatrix} & \cdots & \mathcal{B}^{0,N_s-1} \\ \vdots & \ddots & \vdots & & \vdots \\ \begin{smallmatrix}0&0&0\\0&0&0\\0&0&0\end{smallmatrix} & \cdots & \begin{smallmatrix}1&0&0\\0&1&0\\0&0&1\end{smallmatrix} & \cdots & \begin{smallmatrix}0&0&0\\0&0&0\\0&0&0\end{smallmatrix} \\ \vdots & & \vdots & \ddots & \vdots \\ \mathcal{B}^{N_s-1,0} & \cdots & \begin{smallmatrix}0&0&0\\0&0&0\\0&0&0\end{smallmatrix} & \cdots & \mathcal{B}^{N_s-1,N_s-1} \end{bmatrix} \begin{bmatrix} \mathbf{U}_0 \\ \vdots \\ \mathbf{U}_p \\ \vdots \\ \mathbf{U}_{N_s-1} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_0 \boxed{-\mathcal{B}^{0,p}\,\mathbf{U}_p^*} \\ \vdots \\ \boxed{\mathbf{U}_p^*} \\ \vdots \\ \mathbf{F}_{N_s-1} \boxed{-\mathcal{B}^{N_s-1,p}\,\mathbf{U}_p^*} \end{bmatrix} \tag{2.16}$$

Il est intéressant de remarquer que dans le cas où le déplacement imposé est nul, c'est-à-dire lorsque l'on veut fixer un point, il suffit de retirer la ligne et la colonne qui lui correspondent dans le système linéaire.

Si cette méthode est souvent utilisée en pratique, elle n'est pas correcte d'un point de vue mathématique. En effet, en modifiant ainsi la matrice de rigidité et le second membre, on change la dimension de l'espace vectoriel dans lequel on résout le système d'équations, et on ne garantit plus que la solution finale corresponde à un minimum global. La bonne méthode (en théorie), consiste à utiliser les multiplicateurs de Lagrange. Ainsi, pour imposer des déplacements donnés à un ensemble de r sommets :

$$\mathbf{U}_p = \mathbf{U}_p^* , \tag{2.17}$$

on modifie le système linéaire $[K]\mathbf{U} = \mathbf{F}$, pour prendre en compte les nouvelles contraintes. On se ramène alors à résoudre le système :

$$\left[\begin{array}{c|c} K & \overline{K} \\ \hline \overline{K}^t & 0 \end{array}\right] \left[\begin{array}{c} \mathbf{U} \\ \hline \boldsymbol{\lambda} \end{array}\right] = \left[\begin{array}{c} \mathbf{F} \\ \hline \mathbf{U}^* \end{array}\right] \tag{2.18}$$

où $\boldsymbol{\lambda}$ est le vecteur des multiplicateurs de Lagrange et $\overline{K}$ est la matrice qui contient les $3*r$ vecteurs de base correspondant aux sommets qui sont contraints (Zienkiewicz, 1977; Cotin, 1997; Legat, 2000).

Une fois les conditions aux limites imposées, on peut procéder à la construction de la base de déformation. Pour cela on considère un sommet $\mathbf{P}_l$ du maillage. On lui applique une force élémentaire $\mathbf{F}_x^e$ dans la direction x (en modifiant le second membre du système, comme nous venons de le voir) et on résout le système linéaire. On obtient donc les déplacements de tous les nœuds du maillage engendrés par l'application de cette force élémentaire. Alors, pour chaque sommet du maillage $\mathbf{P}_m$, on stocke ce déplacement $\mathbf{U}_m^x = [u_m^x \;\; v_m^x \;\; w_m^x]^t$ dans la première colonne du tenseur T^{lm}. On réitère l'opération avec des forces élémentaires dans les directions y et z, ce qui permet de trouver la deuxième et la troisième colonne du tenseur T^{lm}. Une fois que l'on a fait cette opération pour tous les sommets du maillage, on sauvegarde dans un fichier l'ensemble des ${N_s}^2$ tenseurs T^{lm} représentant les déplacements subis par le sommet $\mathbf{P}_m$ lorsque le sommet $\mathbf{P}_l$ est soumis à des forces élémentaires dans les trois directions de l'espace :

$$\begin{matrix} \mathbf{F}_x^e & \mathbf{F}_y^e & \mathbf{F}_x^e \\ \downarrow & \downarrow & \downarrow \end{matrix} \tag{2.19}$$

$$T^{lm} = \begin{bmatrix} u_m^x & u_m^y & u_m^z \\ v_m^x & v_m^y & v_m^z \\ w_m^x & w_m^y & w_m^z \end{bmatrix} \tag{2.20}$$

! Remarques :

❶ Le même raisonnement peut être utilisé en imposant des déplacements élémentaires à la place des forces (c'est le cas par exemple dans les travaux de Cotin *et al* (Cotin, 1997; Cotin et al., 1999)). En fait tout dépend de la façon dont on va ensuite utiliser les pré-calculs : si on veut principalement appliquer des forces, on a tout intérêt à pré-calculer le modèle en forces, et inversement. Nous pensons, par exemple, que les pré-calculs en force que nous proposons aurait un grand intérêt pour un modèle de type hybride comme celui proposé par Cotin (Cotin et al., 2000).

❷ Le plus souvent, on peut se contenter de ne pré-calculer que les sommets de la surface du maillage. En effet, le déplacement des sommets situés à l'intérieur ne présente que peu d'intérêt, à moins de vouloir visualiser la déformation de certaines zones spécifiques situées à l'intérieur de l'organe (par exemple les vaisseaux dans le cas du foie). Mais il faut souligner que, même dans le cas où on ne pré-calcule que les forces appliquées aux sommets de la surface, les déformations, elles, correspondent bien à un comportement **volumique**.

❸ Les pré-calculs correspondent en fait à une inversion de la matrice de rigidité K. La différence est que l'on ne calcule qu'une seule colonne de l'inverse de K à chaque fois. Cette méthode peut donc s'appliquer à des systèmes de taille plus grande, pour lesquels l'inversion directe de la matrice dépasserait largement les capacités de l'ordinateur. Une autre méthode permet de diminuer la taille de la matrice de rigidité du système : la *condensation*. Le principe de cette méthode est de redistribuer la contribution des sommets situés à l'intérieur du corps déformable vers les sommets de la surface (Kardestuncer, 1987; Bro-Nielsen et Cotin, 1996). La matrice résultant de cette opération est plus petite, mais elle est aussi plus dense.

❹ Il est intéressant de remarquer que $[T^{ml}] = [T^{lm}]^t$ (ce qui revient à dire que l'inverse de K est symétrique, cf. remarque ❸). On peut donc se contenter de ne stocker que la moitié des pré-calculs.

❺ Si le problème que l'on se pose possède une *formulation intégrale* sur sa frontière, on peut alors utiliser la méthode des éléments finis de frontière (*Boundary Element Mehtod*), qui est très populaire en électromagnétisme, mais que l'on retrouve parfois dans le domaine de l'animation (James et Pai, 1999). Le principal avantage est de réduire ainsi la complexité du problème en passant d'une formulation volumique à son équivalent surfacique (qui est, en revanche, plus dense). Par contre, seul un nombre limité de problèmes élastiques (même linéaires) se prêtent à cette méthode (Hunter et Pullan, 1998).

2.4.1.2 Utilisation de la base de déformations

L'utilisation de la base de déformations met en œuvre des algorithmes différents suivant les conditions aux bords que l'on veut appliquer au modèle. Nous allons donc envisager les différents cas, en allant du plus simple au plus compliqué :

- **Application d'une force en un seul point** :
  C'est le cas le plus simple. Supposons que l'on veuille appliquer une force $\mathbf{F}_m$ sur le sommet $\mathbf{P}_m$. On commence par exprimer la force $\mathbf{F}_m$ dans la base définie par les forces élémentaires $\mathbf{F}^e$ utilisées lors des pré-calculs :

$$\mathbf{F}_m^{\mathcal{B}_e} = \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} \quad \text{où} \quad \mathbf{F}_m = \begin{bmatrix} \alpha\ \mathbf{F}_x^e \\ \beta\ \mathbf{F}_y^e \\ \gamma\ \mathbf{F}_z^e \end{bmatrix} \tag{2.21}$$

! Remarque : Si on a choisi des forces élémentaires de norme 1, cette première étape est inutile.

Alors, chaque sommet $\mathbf{P}_l$ du maillage est soumis au déplacement $\mathbf{U}_l$ défini par :

$$\boxed{\mathbf{U}_l = \left[T^{lm}\right]\ \mathbf{F}_m^{\mathcal{B}_e}} \tag{2.22}$$

- **Application de forces en plusieurs points** :
  On applique exactement le même raisonnement que pour le cas précédent, mais avec plusieurs points d'application des forces. Le déplacement de chaque point du maillage est la somme des déplacements engendrés par chacune des forces. Si on nomme S l'ensemble des indices des sommets sur lesquels une force est appliquée, on peut écrire :

$$\boxed{\mathbf{U}_l = \sum_{m \in \mathcal{S}} \left[T^{lm}\right]\ \mathbf{F}_m^{\mathcal{B}_e}} \tag{2.23}$$

- **Application d'un déplacement en un seul point** :
  Pour pouvoir utiliser la base de déformations, il faut se ramener à l'application d'une force. Il

va donc falloir calculer la force $\mathbf{F}_m$ qui entraînerait le déplacement $\mathbf{U}_m$ que l'on veut appliquer au sommet $\mathbf{P}_m$. En appliquant la formule 2.22 au déplacement $\mathbf{U}_m$, on peut écrire :

$$\mathbf{U}_m = \left[T^{mm}\right] \mathbf{F}_m^{\mathcal{B}_e} \iff \mathbf{F}_m^{\mathcal{B}_e} = \left[T^{mm}\right]^{-1} \mathbf{U}_m \tag{2.24}$$

Il ne reste plus alors qu'à retrouver les déplacements de tous les noeuds $\mathbf{P}_l$ en appliquant la force $\mathbf{F}_m^{\mathcal{B}_e}$ (équation 2.22) :

$$\boxed{\mathbf{U}_l = \left[T^{lm}\right] \left[T^{mm}\right]^{-1} \mathbf{U}_m} \tag{2.25}$$

- **Application de déplacements en plusieurs points** :
  C'est dans ce cas que les choses se compliquent un peu. En effet, le principe de superposition ne fonctionne plus aussi bien si on raisonne en déplacements. Si on veut imposer des déplacements en plusieurs noeuds, il faut tenir compte des interactions entre les noeuds à déplacer. On peut voir, sur la figure 2.5, le problème qui se pose avec des déplacements imposés sur deux sommets. On voit que si on calcule les forces à appliquer sur les sommets comme dans le cas

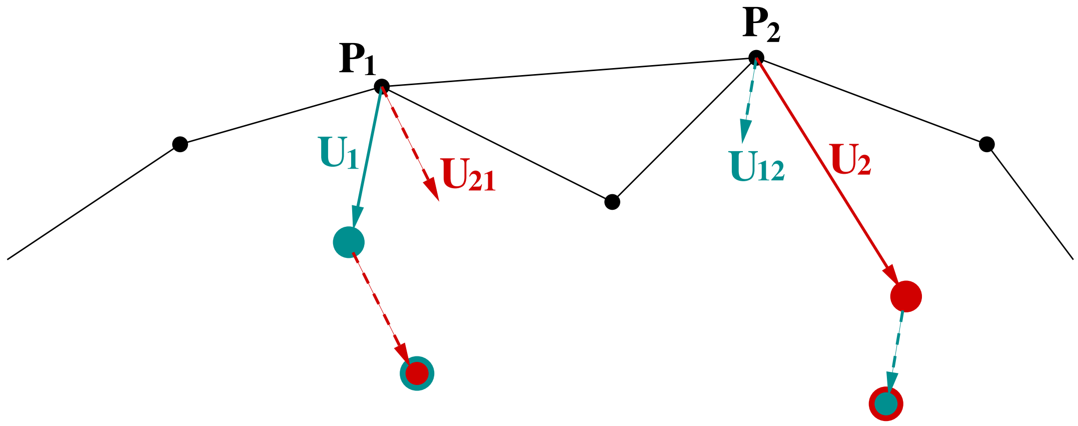


FIG. 2.5 – *Problème posé par l'application de déplacements sur deux sommets simultanément*

précédent, le sommet $\mathbf{P}_1$ se déplace de $\mathbf{U}_1$, mais est aussi soumis à l'influence du déplacement du sommet $\mathbf{P}_2$. Cette influence est caractérise par $\mathbf{U}_{21}$, le déplacement imposé par $\mathbf{P}_2$ sur $\mathbf{P}_1$. Résultat, ces deux sommets se sont trop déplacés. Pour trouver les forces $\mathbf{F}_1^{\mathcal{B}_e}$ et $\mathbf{F}_2^{\mathcal{B}_e}$ qu'il faut réellement appliquer aux deux sommets pour obtenir les déplacements $\mathbf{U}_1$ et $\mathbf{U}_2$, il faut résoudre le système suivant :

$$\left\{ \begin{array}{ll} \left[T^{11}\right] \mathbf{F}_1^{\mathcal{B}_e} + \underbrace{\left[T^{12}\right] \mathbf{F}_2^{\mathcal{B}_e}}_{\mathbf{U}_{21}} & = \mathbf{U}_1 \\ \underbrace{\left[T^{21}\right] \mathbf{F}_1^{\mathcal{B}_e}}_{\mathbf{U}_{12}} + \left[T^{22}\right] \mathbf{F}_2^{\mathcal{B}_e} & = \mathbf{U}_2 \end{array} \right. \tag{2.26}$$

De manière plus générale, si on veut appliquer sur n sommets des déplacements $\mathbf{U}_m$ ($m = 0..n-1$), on peut écrire ce système sous forme matricielle :

$$\begin{bmatrix} [T^{0,0}] & [T^{0,1}] & \cdots & [T^{0,n-1}] \\ [T^{1,0}] & [T^{1,1}] & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ [T^{n-1,0}] & \cdots & \cdots & [T^{n-1,n-1}] \end{bmatrix} \begin{bmatrix} \mathbf{F}_0^{\mathcal{B}_e} \\ \vdots \\ \vdots \\ \mathbf{F}_{n-1}^{\mathcal{B}_e} \end{bmatrix} = \begin{bmatrix} \mathbf{U}_0 \\ \vdots \\ \vdots \\ \mathbf{U}_{n-1} \end{bmatrix} \tag{2.27}$$

En résolvant ce système, on trouve les forces $\mathbf{F}_m^{\mathcal{B}_e}$ ($m = 0..n-1$) qu'il faut appliquer sur les sommets $\mathbf{P}_m$ (On utilise ici aussi pour résoudre le système l'algorithme de gradient conjugé pré-conditionné de la *"Matrix Template Library"* (Lumsdaine et Siek, 1998; Lumsdaine et al., 1998)).
Le déplacement d'un sommet $\mathbf{P}_l$ du maillage est donc finalement :

$$\boxed{\mathbf{U}_l = \sum_{m=0}^{n-1} \left[T^{lm}\right] \ \mathbf{F}_m^{\mathcal{B}_e}} \tag{2.28}$$

La figure 2.6 illustre ce cas. On veut appliquer un même déplacement aux trois sommets de la face selectionnée. Le maillage en rendu de surface représente la position au repos et le maillage en fil de fer la position déformée. On voit alors les forces qu'il a effectivement fallu appliquer pour obtenir les déplacements souhaités :

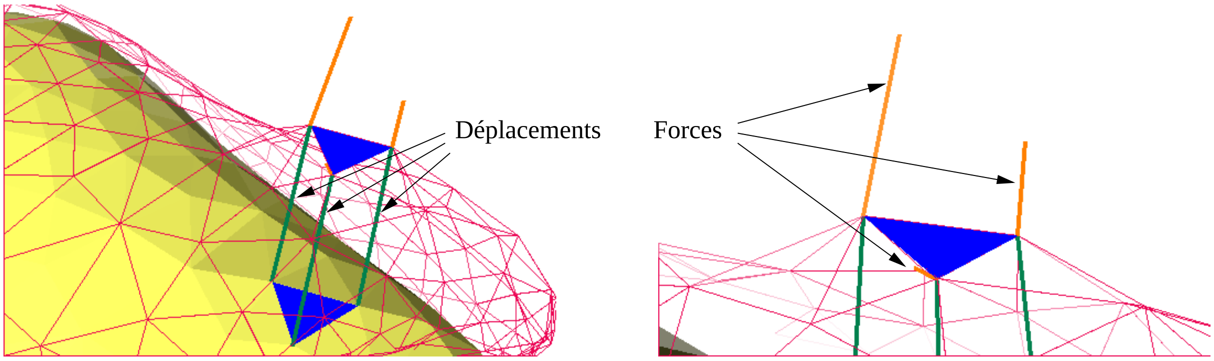
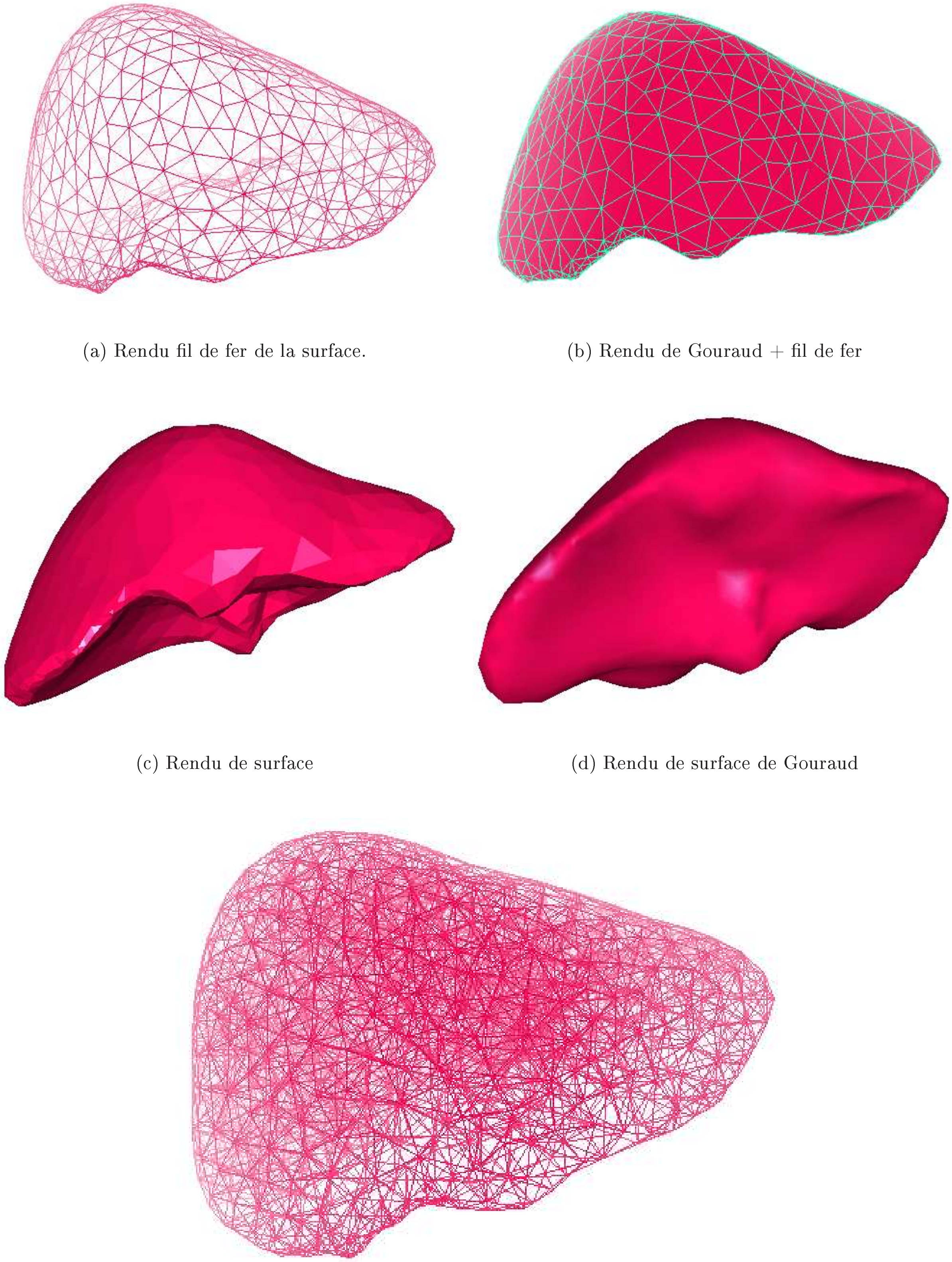


FIG. 2.6 – *Application d'un même déplacement aux trois sommets de la face bleue. On voit en orange les forces qu'il faut appliquer sur ces trois sommets pour obtenir les déplacements souhaités (en vert).*

2.4.1.3 Résultats : performances et exemples

Le maillage du foie que nous utilisons contient 1394 sommets et 8347 arêtes pour 6342 tétraèdres (figure 2.7). Sa surface est composée de 1224 triangles (figure 2.7(a) et (b)), ce qui permet d'obtenir une représentation relativement lisse dès lors que l'on utilise des techniques de rendu comme le *Gouraud Shading* (figure 2.7(d)). Ce maillage est fixé par un certain nombre de sommets sur sa



(a) Rendu fil de fer de la surface.

(b) Rendu de Gouraud + fil de fer

(c) Rendu de surface

(d) Rendu de surface de Gouraud

(e) Rendu fil de fer du volume

FIG. 2.7 – *Les différents modes de visualisation du maillage tétraédrique du foie.*

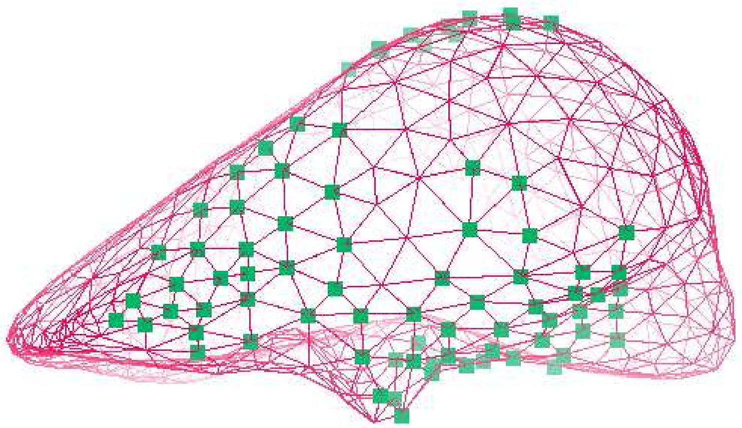


FIG. 2.8 – *Points fixés sur la face antérieure du foie*

face antérieure (figure 2.8), le reste de l'organe étant libre. Dans la mesure où ce modèle déformable subira uniquement des contraintes appliquées sur sa surface (forces de pression ou de contact), nous n'effectuons les pré-calculs que pour les sommets situés sur la surface du maillage, ce qui représente 614 points. Le temps de calcul nécessaire à la construction de la base de déformation va dépendre énormément des coefficients d'élasticité. En effet, pour simuler un matériau quasiment incompressible (le foie est constitué presque exclusivement de liquide), le coefficient λ doit tendre vers l'infini. Dans la pratique, on essaiera toujours de le prendre le plus grand possible. Mais si il est trop grand, la matrice de rigidité K devient très mal conditionnée et l'algorithme de gradient conjugué a du mal à converger (le pré-conditionnement de la matrice est alors une étape indispensable). On peut toutefois donner un ordre d'idée : pour des valeurs $\lambda = 31034\ kg/cm^2$ et $\mu = 3448\ kg/cm^2$ (qui correspondent à $E = 10000\ kg/cm^2$ et $\nu = 0.45$, voir équations 1.27), les pré-calculs demandent un peu plus de 4 heures de calcul. La taille du fichier de stockage des données est de 36 Megabits.

Le fait que le modèle pré-calculé soit ramené à un modèle surfacique, ajouté à la simplicité d'utilisation des pré-calculs, permet de réduire au minimum la structure de données nécessaire aux stockage et à la simulation. En pratique, on peut se contenter d'un maillage composé d'un ensemble de sommets et d'un ensemble de faces. Les sommets ont une position au repos, une position déformée et un tableau de tenseurs permettant de calculer les déplacements qu'ils vont imposer à tous les sommets du maillage lorsqu'on leur appliquera une force. Les faces connaissent leurs trois sommets, et servent presque exclusivement à représenter la surface du maillage et à calculer des normales.

Le tableau 2.1 regroupe les performances atteintes par le modèle déformable pré-calculé lors de l'application de différentes contraintes aux bords. On constate que ce modèle permet largement de garantir un fonctionnement en temps réel ($>$25 Hz), même dans le cas le moins adapté où l'on veut imposer des déplacements en un grand nombre de nœuds.

La figure 2.9 montre les résultats obtenus avec ce modèle pré-calculé lors de l'application de différentes forces sur un sommet du maillage. En plus de son efficacité, ce modèle fournit une

Fréquences de simulation (modèle de foie dont la surface est composée de 614 sommets)		Pentium PII 450 MHz	Pentium PIII 600 MHz	SGI Onyx2 IR 195 MHz
Forces sur 1 sommet		3120 Hz	3772 Hz	3846 Hz
Déplacements sur	1 sommet	3081 Hz	3759 Hz	3745 Hz
	5 sommets	419 Hz	561 Hz	510 Hz
	10 sommets	142 Hz	185 Hz	167 Hz
	20 sommets	30 Hz	40 Hz	36 Hz

TAB. 2.1 – *Performances du calcul de déformations pour l'application de différents types de contraintes. Dans le cas de l'application de forces, on ne donne les performances que pour 1 force, car le temps de calcul est linéaire en fonction du nombre de forces appliquées.*

déformation d'une grande précision (qui dépend néanmoins de la finesse et de la qualité du maillage), comme en atteste la régularité du champ de déplacements obtenu.

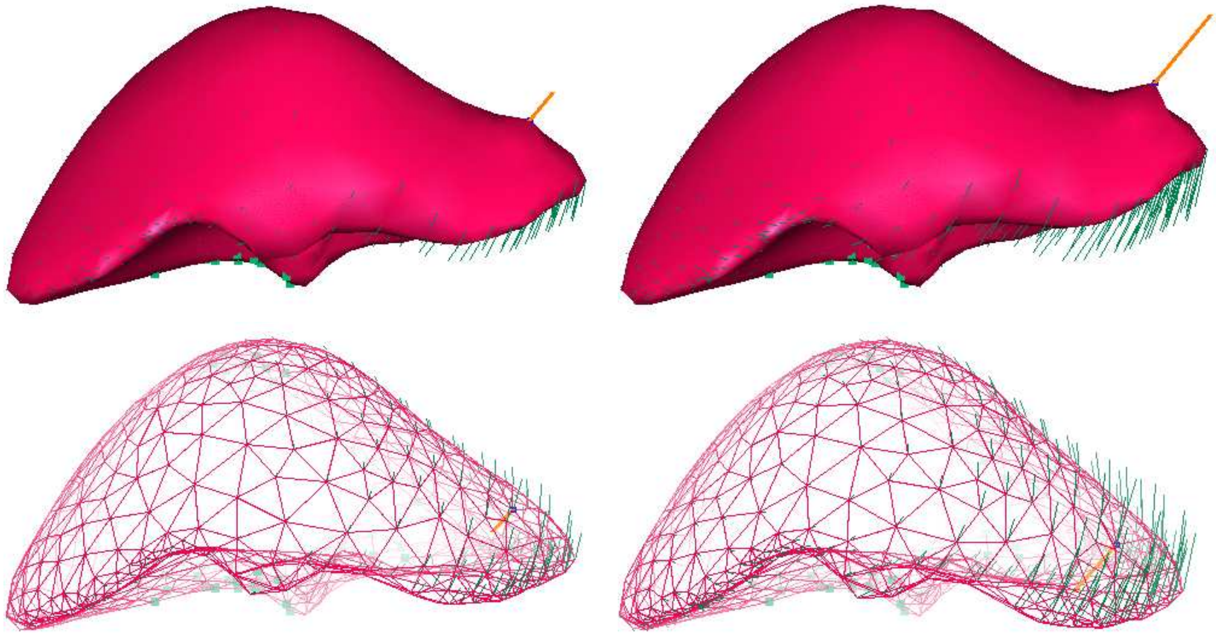


FIG. 2.9 – *Déformations du foie en utilisant le modèle pré-calculé. Le trait orange représente la force appliquée sur un sommet, tandis que les lignes vertes permettent de visualiser les déplacements des sommets par rapport à leur position de repos.*

2.4.1.4 Application : couplage avec un rendu réaliste

Dans le cadre de l'AISIM, nous avons été amené à collaborer étroitement avec le projet iMAGIS de l'INRIA Rhône-Alpes (à Grenoble) afin de coupler notre modèle déformable pré-calculé avec un rendu réaliste. En effet, F. Neyret et M.-P. Cani ont mis au point une méthode pour plaquer des textures sur des objets complexes avec très peu de distorsions (Neyret et Cani, 1999). L'utilisation de motifs de base triangulaires permet d'éviter la formation de pôles, comme c'est le cas quand on utilise des motifs carrés pour texturer une surface à géométrie sphérique. De plus, une seconde couche de texture permet de représenter des reflets très réalistes. Parallèlement, P.-O. Agliati a développé une interface permettant de coupler ce rendu texturé et ces reflets avec un modèle déformable (Agliati, 1999). Cette interface gère la détection de collision entre la surface et un instrument manipulé par l'utilisateur et interprète ces collisions en terme de contraintes aux bords à appliquer au modèle déformable. Nous avons donc couplé cette interface au modèle pré-calculé que nous venons de présenter.

La figure 2.10 montre les résultats obtenus, d'abord simplement pour la visualisation de notre modèle de foie, puis lors de la simulation de sa déformation à l'aide d'un outil. On constate que les reflets améliorent énormément le réalisme visuel de ce modèle en apportant un aspect "humide" à la surface (ce qui est vrai pour la plupart des organes internes). De plus, les textures et les reflets aident tous deux à la compréhension de la forme de l'objet, car ils apportent des informations supplémentaires qui seront interprétées par le cerveau pour reconstruire la profondeur. C'est encore plus vrai lorsque l'on déforme le foie, car l'apparition de nouveaux reflets et la distortion locale de la texture permettent de mieux percevoir la déformation.

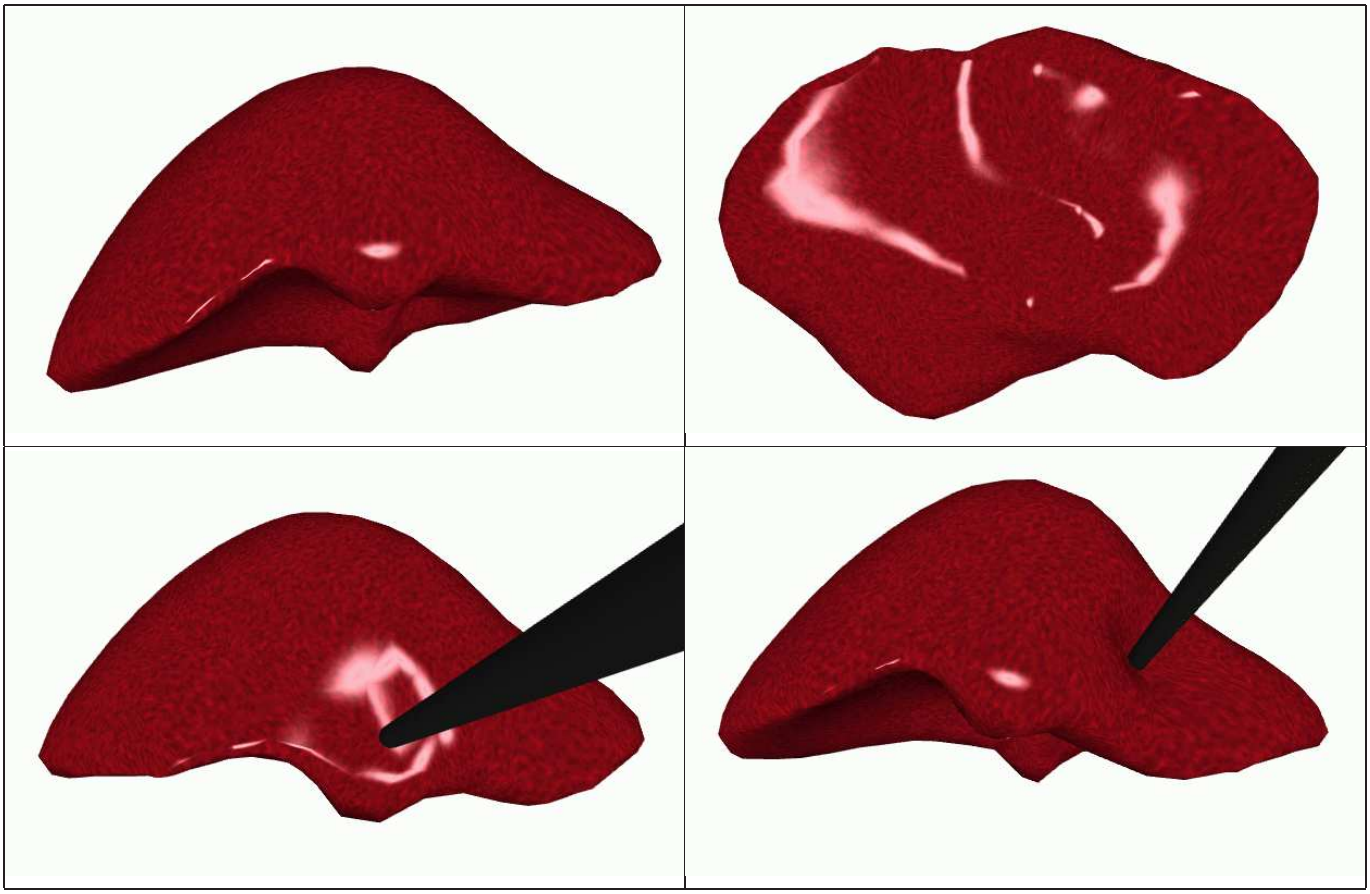


FIG. 2.10 – *Utilisation du modèle déformable pré-calculé couplé avec un rendu réaliste composé d'une texture et de reflets (collaboration avec le projet iMAGIS : F. Neyret (Neyret et Cani, 1999) et P.-O. Agliati (Agliati, 1999))*

2.4.1.5 Bilan

Ce premier modèle déformable permet donc de calculer avec une grande efficacité et une grande précision les déformations d'objets à la géométrie relativement complexe. Mais ces performances sont atteintes au prix d'une restriction assez pénalisante en vue d'une application à la simulation de chirurgie : il est impossible de procéder à des changements de topologie du maillage. En effet, toute modification de la structure du modèle entraîne une remise à jour de la matrice de rigidité globale du système. Or cette remise à jour impose de recommencer les pré-calculs à chaque changement intervenant dans le maillage. Il est donc, en particulier, impossible de simuler une découpe, qui est un des gestes élémentaires effectué par les chirurgiens lors d'interventions chirurgicales.

Pour cette raison, nous proposons un autre modèle déformable basé sur les mêmes théories de l'élasticité linéaire et des éléments finis, et utilisant les mêmes équations pour calculer les forces exercées en chaque nœud du maillage (équation 2.13), mais dont le schéma de résolution et la structure de données adaptée permettent, à tout instant, de modifier de manière très efficace la topologie du maillage.

2.4.2 Résolution explicite dynamique : modèle masse-tenseur

Contrairement au modèle pré-calculé présenté au paragraphe précédent, le modèle masse-tenseur ne nécessite pas l'assemblage de la matrice globale de rigidité. En effet, toutes les données nécessaires au calcul des forces sont directement stockées dans la structure de données liée au maillage. À partir du maillage tétraédrique de l'objet que l'on veut simuler – dans notre cas une structure anatomique – on construit une structure de données contenant les notions de sommets, d'arêtes et de tétraèdres. Pour chaque sommet, nous stockons la liste de ses tétraèdres voisins ainsi que celle de ses arêtes adjacentes, sa position au repos $\mathbf{P}_p^0$, sa position courante $\mathbf{P}_p$ et son tenseur de rigidité $\mathcal{B}^{pp}$ (équation 2.12). Chaque arête connaît ses deux sommets voisins ainsi que la liste des tétraèdres qui lui sont adjacents, et contient son tenseur de rigidité $\mathcal{B}^{pj}$. Enfin pour chaque tétraèdre, nous stockons ses quatre sommets, ses six arêtes, les constantes l'élasticité λ et μ et les quatre vecteurs de forme $\boldsymbol{\alpha}_k$. Il est aussi intéressant d'avoir une notion de triangle pour représenter la surface du maillage. Dans ce cas, chaque face externe connaîtra le tétraèdre auquel elle appartient et ses trois sommets. Cette notion de triangulation de surface permet en particulier de calculer de manière plus efficace les normales à la surface qui seront utiles pour afficher l'objet. Elles serviront aussi lors de la modélisation des contacts avec les outils chirurgicaux (chapitre 4).

Le principe de fonctionnement du modèle masse-tenseur est relativement simple. À chaque instant t, on applique éventuellement de nouvelles conditions au bords, puis on calcule les forces élastiques exercées sur chacun des sommets du maillage en utilisant l'équation 2.13. On intègre alors, pour chaque sommet, les équations du mouvement pour trouver sa nouvelle position à l'instant $t+1$.

2.4.2.1 Les condititions aux limites

Comme pour le modèle précédent, il faut prévoir un ensemble de conditions aux bords qui permettront de fixer ce modèle ou d'interagir avec lui :

- On peut fixer un sommet. Il suffit pour cela de ne pas tenir compte des forces externes et internes qu'il subit et de ne jamais calculer sa nouvelle position.
- Pour pré-contraindre le modèle, on peut aussi déplacer un sommet avant de le fixer.
- On peut appliquer un déplacement sur un ou plusieurs sommets.
- On peut aussi appliquer des forces que l'on ajoute simplement aux forces internes avant d'intégrer les équations du mouvement. On distingue alors plusieurs types de forces : les forces

instantanées qui ne durent que le temps d'une itération, les forces permanentes qui sont réappliquées à chaque itération, et les champs de forces qui correspondent à des forces permanentes s'appliquant sur l'ensemble du maillage.

2.4.2.2 Le calcul des forces

Afin de minimiser le nombre d'opérations, on n'applique pas directement la formule 2.13. On commence par parcourir les arêtes du maillage. Pour chaque arête ($\mathbf{P}_p$,$\mathbf{P}_j$), on calcule la force qu'elle exerce sur ses deux sommets : $[\mathcal{B}^{pj}] * \mathbf{U}_j$ pour le sommet $\mathbf{P}_p$ et $[\mathcal{B}^{pj}]^t * \mathbf{U}_p$ pour le sommet $\mathbf{P}_j$. Ensuite, on parcourt l'ensemble des sommets. Pour chaque sommet $\mathbf{P}_p$, on calcule la force due à son déplacement : $[\mathcal{B}^{pp}] * \mathbf{U}_p$. On peut alors intégrer l'équation du mouvement pour trouver la nouvelle position du sommet.

On remarque que ce schéma de fonctionnement est très proche de celui d'un réseau masse-ressort. En effet, lorsque l'on utilise un modèle masse-ressort, on parcourt d'abord chaque arête ($\mathbf{P}_p$,$\mathbf{P}_j$) du maillage (qui modélise un ressort) pour calculer les forces qu'elle exerce sur ses deux sommets :

$$\mathbf{F}_p^{spring} = -\mathbf{F}_j^{spring} = k\left(\frac{l-l_0}{l}\right)\overrightarrow{\mathbf{P}_p\mathbf{P}_j} \quad \text{où} \quad \begin{cases} l_0 & \text{est la longueur du ressort au repos} \\ l & \text{est la longueur du ressort déformé} \\ k & \text{est la raideur du ressort} \end{cases} \cdot \tag{2.29}$$

Ensuite, il n'y a plus qu'à parcourir les sommets pour intégrer le mouvement et obtenir les nouvelles positions des sommets.

2.4.2.3 L'intégration du mouvement

Nous considérons que l'équation qui gouverne le mouvement de notre modèle élastique linéaire est une équation différentielle Newtonienne de la forme :

$$m_i\frac{d^2\mathbf{P}_i}{dt^2} = \gamma_i\frac{d\mathbf{P}_i}{dt} + \mathbf{F}_i \tag{2.30}$$

Cette équation est reliée aux équations différentielles rencontrées en mécanique des milieux continus (Bathe, 1982) :

$$M\ddot{\mathbf{U}} + C\dot{\mathbf{U}} + K\mathbf{U} = \mathbf{R} \tag{2.31}$$

D'après la théorie des éléments finis, la matrice de masse M et la matrice d'amortissement C sont creuses et dépendent des propriétés de chaque tétraèdre. Dans notre cas, nous considérons que M et C sont des matrices diagonales, c'est-à-dire que la masse et les effets d'amortissement sont concentrés sur les sommets du maillage. Cette simplification, appelée *masse-lumping*, découple les mouvements de chacun des nœuds et permet donc d'écrire l'équation 2.31 comme un ensemble d'équations différentielles indépendantes du type 2.30, une pour chaque sommet.

De plus, nous avons choisi un schéma d'intégration explicite, dans lequel les forces élastiques sont estimées à l'instant t pour calculer la nouvelle position du sommet à l'instant $t+1$:

$$\left(\frac{m_i}{\Delta t^2} - \frac{\gamma_i}{2\Delta t}\right)\mathbf{P}_i^{t+1} = \mathbf{F}_i + \frac{2m_i}{\Delta t^2}\mathbf{P}_i^t - \left(\frac{m_i}{\Delta t^2} + \frac{\gamma_i}{2\Delta t}\right)\mathbf{P}_i^{t-1} \tag{2.32}$$

Les schémas explicites sont seulement *conditionnellement stables*, ils convergent donc moins vite que les schémas implicites. Pour que le schéma soit stable, le pas de temps utilisé pour l'intégration numérique doit être suffisamment petit. Le pas de temps critique dépend de la plus grande valeur propre de la matrice de rigidité et des valeurs locales de la masse et de l'amortissement (Bathe, 1982). Pour optimiser le pas de temps, nous prenons la matrice de masse égale à l'identité, et nous cherchons à optimiser les valeurs locales de l'amortissement afin de limiter les risques d'oscillations. Ces opérations ont pour effet d'améliorer le conditionnement du système linéaire, et donc d'en stabiliser la résolution.
Voici les différentes valeurs de l'amortissement que nous avons essayées :

❶ Le plus simple est de prendre la matrice C égale à une constante fois l'identité. Cela revient à considérer une même valeur pour tous les sommets du maillage. On a alors affaire à un amortissement isotrope et homogène.

❷ On peut faire un peu mieux en prenant C diagonale, mais avec chaque sous bloc (3x3) représentant le sommet $\mathbf{P}_i$ valant $c_i * Id$. On obtient ainsi une valeur différente de l'amortissement pour chaque sommet. Il est alors intéressant de prendre l'amortissement proportionnel à la raideur locale (on pourra prendre par exemple c_i proportionnel à la norme du tenseur de rigidité du sommet). L'amortissement devient dans ce cas non homogène, ce qui revient à amortir plus fortement les zones où la rigidité est plus importante.

❸ Dans les deux cas précédents, on ne considérait qu'une valeur scalaire par sommet, ce qui implique un amortissement isotrope. On peut améliorer les résultats en prenant un tenseur d'amortissement pour chaque sommet, ce qui revient à prendre C bloc-diagonale, c'est-à-dire dont les seuls éléments non nuls sont des blocs (3x3) disposés sur la diagonale. Il est alors classique de prendre ces tenseurs d'amortissement proportionnels au tenseur de rigidité ou à sa diagonale (pré-conditionnement de Jacobi physique). L'amortissement se fera alors prioritairement dans la direction de plus forte rigidité. Si un tel amortissement apporte une amélioration dans le cas d'un matériau isotrope (il tient parfaitement compte du comportement local du matériau), son intérêt est décuplé lorsque l'on considère un matériau anisotrope, comme ce sera le cas dans la seconde partie de ce chapitre (paragraphe 2.5).

❹ Jusque là, nous n'avons considéré que des effets d'amortissement concentrés sur les sommets. C'est d'ailleurs ce qui nous a permis d'écrire l'équation 2.32 en dissociant le mouvement de chaque sommet. On peut interpréter ce comportement en imaginant que chaque sommet est plongé dans un liquide visqueux qui dissipe une partie de l'énergie sous forme de frottements. Mais en fait dans un matériau structuré comme les matériaux élastiques, le principal amortissement est du aux cisaillements créés par les différences de vitesses des sommets voisins. Or on ne tient pas du tout compte de cette effet, qui devrait agir comme un lissage spatial du champ de vitesses. Mais le calcul de cet amortissement fait intervenir les sommets voisins au sommet considéré, ce qui ne permet plus de dissocier les équations du mouvement.

   On peut contourner ce problème en intégrant le calcul de l'amortissement à celui de la force par l'intermédiaire du *strain rate tensor* (tenseur du taux de déformation) (O'Brien et Hodgins, 1999; Bro-Nielsen, 1996);

$$D = \frac{1}{2}\left[\nabla \mathbf{V}^t + \nabla \mathbf{V}\right], \text{ où } \mathbf{V} \text{ est le champ de vitesse.} \tag{2.33}$$

En utilisant ce nouveau tenseur à la place de E dans la loi de Hooke et en discrétisant ces forces d'amortissement exactement de la même manière que les forces élastiques, on obtient une nouvelle expression de la force élastique amortie exercée sur un sommet du maillage :

$$\boxed{\mathbf{F}_p^l = \left[\mathcal{B}^{pp}\right]\Big(\mathbf{U}_p + \gamma\ \mathbf{V}_p\Big) + \sum_{\mathcal{A}\,\in\,\nu(\mathbf{P}_p)} \left[\mathcal{B}^{pj}\right]\Big(\mathbf{U}_j + \gamma\ \mathbf{V}_j\Big)}, \tag{2.34}$$

où γ est le cœfficient de proportionnalité entre la rigidité et l'amortissement. On peut remarquer que la modification qui intervient sur le premier terme, qui ne concerne que le sommet, correspond exactement à l'amortissement décrit au point ❸.

Trucs et astuces : Lors de tous ces essais pour améliorer la stabilité et la vitesse de convergence du modèle, nous avons été amenés à essayer quelques "bricolages". Un de ceux qui ont donné des résultats intéressants, et que nous continuons à utiliser, est de placer un seuil sur le déplacement des sommets. En effet, nous sommes partis de la constatation que les instabilités du modèle étaient souvent provoquées par une contrainte trop forte mais très localisée (utilisateur ayant "la main lourde" lors de la manipulation, ou bien lors de la découpe du modèle lorsqu'il est sous tension) qui entraînait une vitesse très importante de certains sommets. Une simple borne sur le déplacement des sommets suffit alors à éviter la propagation de l'instabilité. Cette astuce présente en outre l'avantage d'éviter "l'explosion" du maillage qui peut arriver lorsque l'amortissement est insuffisant par rapport à la rigidité du matériau et au pas de temps. Au lieu d'exploser, le modèle se met alors à osciller autour de sa position de repos. Même si cette oscillation n'est pas du tout réaliste, elle donne le temps à l'utilisateur de mieux régler les paramètres au lieu de devoir tout réinitialiser. En pratique, nous utilisons un seuil qui correspond à une vitesse instantanée de 0.5 ms^{-1} sans constater la moindre différence visuelle de comportement du modèle (les gestes d'un chirurgien ne dépassent pas quelques centimètres par seconde). Même si cette petite modification n'a aucune réalité physique, elle facilite tellement la vie lors de l'élaboration et du réglage d'un modèle déformable que nous la conseillons sans hésiter.

2.4.2.4 Changements de topologie du maillage

Une des tâches de base, que l'on veut être capable de faire avec un simulateur de chirurgie, consiste à découper les tissus mous. D'autre part, pour des raisons de précision de calcul, ou de finesse de la découpe, on peut être amené à modifier localement la topologie du maillage, par exemple en augmentant le nombre de tétraèdres dans une zone d'intérêt. Sa structure de données et le caractère local du calcul des forces, font du modèle masse-tenseur un outil particulièrement bien adapté pour simuler ce type d'opérations.

Prenons par exemple le cas de la découpe sur un organe tel que le foie. L'importance de sa vascularisation rend l'utilisation d'un scalpel presque impossible en raison du risque de sectionner une grosse veine ou une artère, et de provoquer une hémorragie. La découpe du parenchyme hépatique se fait alors soit avec une pince qui va progressivement écraser le parenchyme qui est relativement friable, soit avec un bistouri à ultrason, qui fera éclater les cellules hépatiques avec lesquelles il est en contact. Dans les deux cas, le chirurgien procède donc par retraits de matière successifs. Il peut ainsi faire apparaître les vaisseaux sanguins sans risque de les endommager. Il ne lui restera alors plus qu'à les clamper avant de les sectionner. Le bon moyen de simuler la découpe sur un modèle du foie semble donc de retirer les tétraèdres qui seront pris dans les mâchoires de la pince ou qui seront en contact avec le bistouri à ultrason. La modélisation des contacts et des gestes chirurgicaux est détaillée au chapitre 4.

Le retrait d'un tétraèdre du modèle masse-tenseur est une opération extrêmement simple. Lors du calcul des tenseurs de rigidité, chaque tétraèdre apporte sa contribution aux tenseurs de rigidité de ses sommets et de ses arêtes (équation 2.12). Il suffit donc, pour annuler le comportement biomécanique du morceau de tissu que représente le tétraèdre, de retirer ses contributions sur ses sommets et ses arêtes (figure 2.11). Cette opération a un coût relativement réduit puisque cela

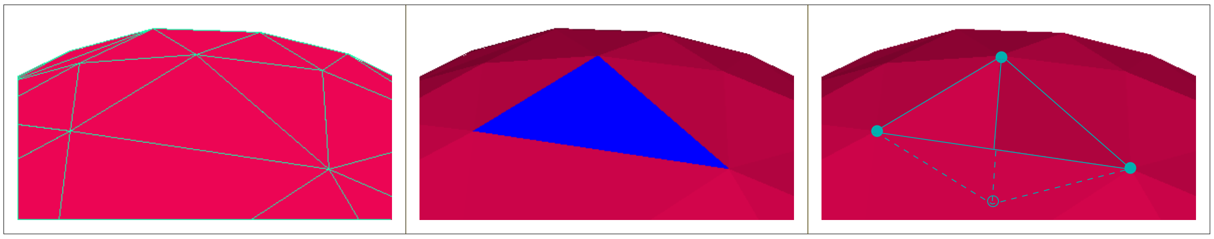


FIG. 2.11 – *Pour retirer le tétraèdre dont la face externe est sélectionnée (en bleu foncé), il suffit de mettre à jours les tenseurs de rigidité sur les sommets et les arêtes adjacents au tétraèdre.*

représente 10 soustractions de matrices 3x3 :

$$\begin{cases} [\mathcal{B}^{jj}] = [\mathcal{B}^{jj}] - [\mathcal{B}^{jj}_{\mathcal{T}}] & \text{pour les 4 sommets } \mathbf{P}_j \text{ du tétraèdre } \mathcal{T}, \\ [\mathcal{B}^{jk}] = [\mathcal{B}^{jk}] - [\mathcal{B}^{jk}_{\mathcal{T}}] & \text{pour les 6 arêtes } (\mathbf{P}_j, \mathbf{P}_k) \text{ du tétraèdre } \mathcal{T}. \end{cases} \tag{2.35}$$

Il faut ensuite modifier la structure du maillage géométrique, en mettant à jour les voisinages des sommets et des arêtes concernés, et le cas échéant, supprimer aussi certains sommets ou certaines arêtes qui se retrouveraient isolés.

En mettant à jour les tenseurs de manière locale, on garantit que le tissu conserve exactement le même comportement que si on avait retiré le tétraèdre sur le maillage au repos. De plus, grâce à la continuité volumique de la modélisation par la méthode des éléments finis, la déformation du tissu reste naturelle et exacte tout au long de la découpe.

De manière tout à fait comparable, il est possible de raffiner localement le maillage. Cela peut permettre d'augmenter le nombre de tétraèdres dans une zone d'intérêt, dans le but d'améliorer la précision des calculs, ou de permettre une découpe plus fine, les tétraèdres supprimés étant alors plus petits. Nous avons par exemple défini sur notre structure de données de maillage une opération de découpe d'une arête en deux en y ajoutant un sommet. Cela revient à couper en deux chaque tétraèdre contenant cette arête comme indiqué sur la figure 2.12. En dehors de la remise à jour du maillage géométrique, il faut modifier les tenseurs de rigidité des sommets et des arêtes qui existaient déjà (en bleu sur la figure 2.12), et calculer ceux du nouveau sommet et des nouvelles arêtes (en jaune sur la figure 2.12). Nous présenterons des résultats de cette méthode de raffinement dans le

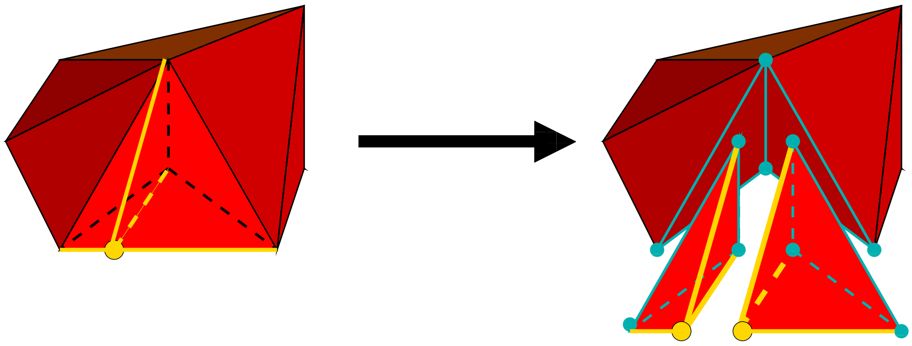


FIG. 2.12 – *Raffinement local du maillage. On coupe une arête en deux en rajoutant un sommet. il faut alors mettre à jour les tenseurs de rigidité des sommets et des arêtes qui existaient déjà (en bleu), et calculer ceux du nouveau sommet et des nouvelles arêtes (en jaune).*

chapitre 4, qui traite de la modélisation des interactions entre les instruments chirurgicaux et les modèles déformables.

2.4.2.5 Performances et exemples

Bien évidement le modèle masse-tenseur est plus long à converger vers une solution que le modèle pré-calculé, pour la simple raison que tous les calculs sont effectués pendant la simulation. Par contre, on atteint assez vite (au bout de quelques itérations seulement), une solution rapprochée. Cette solution sera relativement exacte dans la zone où les contraintes sont appliquées. Par contre la déformation globale du modèle sera plus lente, car les contraintes doivent se propager de proche en proche dans tout le modèle. On évalue donc les performances de ce modèle par le nombre d'itérations que l'on peut faire entre deux affichages, le minimum étant d'en faire une, et donc d'avoir un modèle dont la fréquence de calcul est de 20Hz. Cette limite est atteinte sur un pentium II 450MHz pour un maillage comportant environ 10000 tétraèdres.

Les figures 2.13 et 2.14 présentent les résultats obtenus lors de la suppression de tétraèdres sur des modèles synthétiques précontraints. Pour le cylindre, nous avons fixé les sommets de la face inférieure et avons appliqué des déplacements aux sommets de l'autre face. Nous avons alors supprimé certains tétraèdres, provoquant ainsi un déséquilibre du modèle, qui évolue alors de manière naturelle vers sa nouvelle forme. Nous avons procédé de manière identique sur le deuxième modèle. Les sommets situés sur le bord droit sont fixés et certains sommets du bord gauche ont subi un déplacement. On observe, sur ces deux exemples le comportement réaliste du modèle masse-tenseur lors de changements importants de topologie.

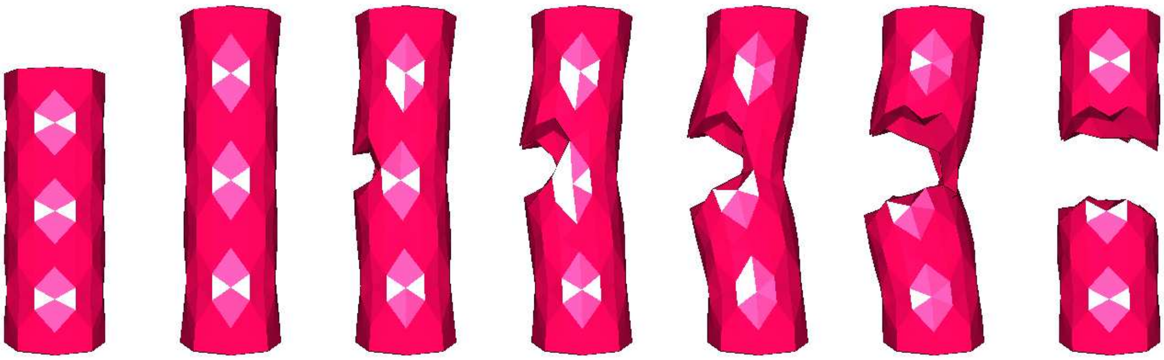


FIG. 2.13 – *Déformation et découpe d'un modèle masse-tenseur représentant un cylindre.*

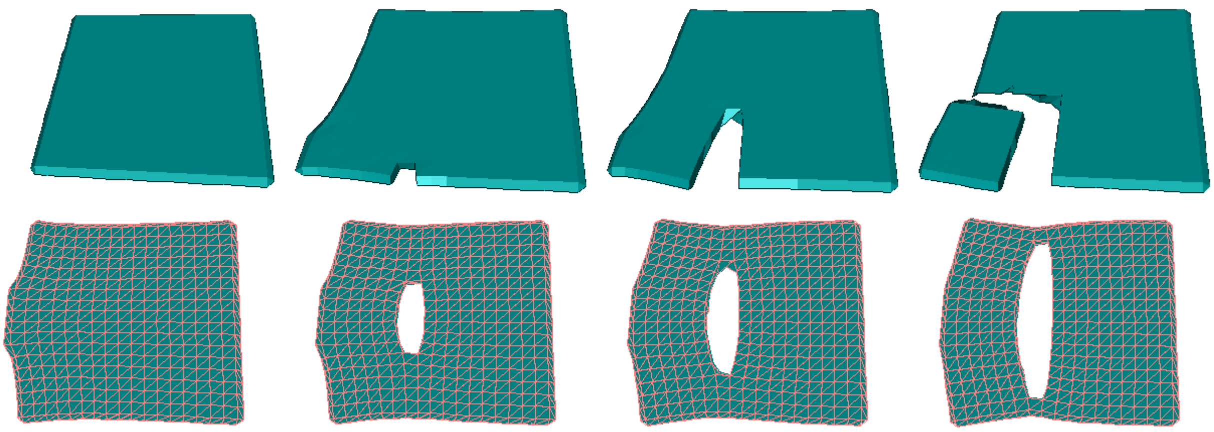


FIG. 2.14 – *Deux exemples de découpe sur un modèle précontraint. La déformation reste réaliste lors de la suppression des tétraèdres.*

2.5 Modélisation de matériaux anisotropes

La loi d'élasticité linéaire que nous avons utilisée jusqu'ici (loi de *Hooke*, équation 2.1) décrit un comportement isotrope. Or la plupart des tissus biologiques présentent de fortes anisotropies. Donc, dés lors que l'on désire modéliser les organes du corps humain, l'utilisation de cette loi isotrope est beaucoup trop restrictive.

Nous présentons dans ce paragraphe les modifications que nous avons apportées à notre modèle d'élasticité linéaire afin d'étendre son comportement biomécanique à celui des matériaux anisotropes. Nous nous sommes particulièrement attachés à modéliser deux comportements anisotropes très répandus dans les structures anatomiques : l'anisotropie provoquée par la présence de fibres et celle due à la présence d'une peau autour des organes.

2.5.1 Matériaux élastiques linéaires "transversalement isotropes"

Beaucoup de structures anatomiques comme les muscles, les tendons, les ligaments ou les vaisseaux sanguins sont fortement anisotropes. Ces tissus, ainsi que de nombreux autres, doivent leur anisotropie à la présence de fibres. Les matériaux qui les composent possèdent alors une direction privilégiée, ou direction de fibre, le long de laquelle la rigidité est plus grande. Dans ce cas, on peut considérer que le comportement du tissu est isotrope dans le plan perpendiculaire à la direction de la fibre. On dit que ces matériaux sont "transversalement isotropes".

Plusieurs ouvrages traitent des développements théoriques liés aux matériaux élastiques transversalement isotropes (Spencer, 1972; Spencer, 1984; Fung, 1993). De plus G. A. Hatfield propose une étude intéressante des lois de conservations qui s'appliquent pour de tels matériaux (Hatfield, 1996; Hatfield, 1997).

On trouve, dans la littérature, plusieurs exemples d'utilisation de ce type d'élasticité pour modéliser des organes du corps humain. M. Kaiss utilise un matériau de St Venant-Kirchhoff transversalement isotrope pour calculer les déformations de l'œil lors d'interventions chirurgicales (Kaiss et Tallec, 1996). J. D. Humphrey *et al* proposent un modèle assez complexe du cœur tenant compte de la présence de fibres, qu'ils valident et dont ils règlent les paramètres en l'identifiant à des données mesurées sur de vrais cœurs (Humphrey et Yin, 1987; Humphrey et al., 1990). Un autre modèle du cœur, linéaire transversalement isotrope, est utilisé par X. Papademetris pour segmenter des séquences temporelles d'images médicales de différents types (IRM, échographie et scanner) (Papademetris et al., 1999; Papademetris, 2000). De nombreux travaux ont aussi été menés sur la modélisation des tendons (en particulier du genou). Le plus souvent l'approche choisie est de compléter un modèle élastique isotrope d'un terme d'énergie pénalisant l'étirement le long de la direction de la fibre. J. A. Weiss *et al* enrichissent ainsi un modèle de Mooney-Rivlin avec un terme d'étirement de fibre et un terme d'interaction entre la fibre et la composante isotrope du matériau. Ils l'utilisent afin de modéliser la déformation d'un ligament collatéral du genou (Weiss et al., 1995; Weiss, 1995; Weiss et al., 1996; Puso et Weiss, 1998). Le même principe peut être appliqué avec un modèle d'élasticité linéaire complété d'une fibre (Martelli et al., 1997). Cependant, aucun de ces modèles n'a pour vocation de respecter des contraintes temps réels.

Par contre, certains auteurs utilisent des modèles masse-ressort pour simuler ce type de comportement en temps-réel. Les réseaux masse-ressort ne pouvant pas garantir l'isotropie du matériau simulé à cause des directions privilégiées que représentent les ressorts, il semble difficile de réellement parler de modèle possédant *une* direction privilégiée. On peut tout de même approximer ce comportement soit en privilégiant une direction dans la géométrie du maillage (Radetzky et al., 1998; Radetzky et al., 2000), soit en ajoutant des ressorts représentant les fibres (Bourguignon et Cani, 2000).

Nous proposons, dans ce paragraphe, de généraliser le modèle élastique linéaire aux matériaux transversalement isotropes. Pour cela, nous définissons des termes d'énergie venant corriger le comportement afin qu'il prenne en compte la présence d'une direction privilégiée de déformation. De plus, nous séparons les termes représentant l'étirement dans la direction d'anisotropie et ceux modélisant le cisaillement entre la direction de la fibre et le plan orthogonal. On pourra ainsi choisir les termes correspondant au comportement recherché.

2.5.1.1 Formulation continue

Nous avons définis, au paragraphe 1.2, les *composantes de déformation* (équations 1.8 et 1.9), qui expriment l'étirement et le cisaillement au sein d'un élément de volume. Si on exprime l'énergie de déformation dans le cas de l'élasticité linéaire isotrope, en fonction de ces composantes de déformation :

$$W^l = \left(\frac{\lambda}{2} + \mu\right)\left({\epsilon^l_x}^2 + {\epsilon^l_y}^2 + {\epsilon^l_z}^2\right) + \lambda\left(\epsilon^l_x\epsilon^l_y + \epsilon^l_x\epsilon^l_z + \epsilon^l_y\epsilon^l_z\right) + \frac{\mu}{2}\left({\gamma^l_{xy}}^2 + {\gamma^l_{xz}}^2 + {\gamma^l_{yz}}^2\right), \qquad (2.36)$$

on vérifie bien qu'elle est isotrope dans la mesure où les termes d'étirement et de cisaillement se voient attribués les mêmes coefficients quelque soit la direction considérée. Imaginons maintenant que l'on veuille augmenter la raideur du matériau dans une direction donnée modélisée par le vecteur unitaire $\mathbf{a}_0$. Il faut alors pénaliser toute déformation dans cette direction, en attribuant à cette direction des coefficients d'élasticité plus grands afin d'augmenter l'énergie de déformation. Pour cela, on définit deux ensembles de coefficients de Lamé :

- (λ^L,μ^L) sont les constantes de Lamé dans la direction privilégiée $\mathbf{a}_0$.
- (λ,μ) sont maintenant les constantes de Lamé dans le plan perpendiculaire à $\mathbf{a}_0$.
- Enfin, pour plus de clarté dans les notations, on pose : $\Delta\lambda = \lambda^L - \lambda$ et $\Delta\mu = \mu^L - \mu$.

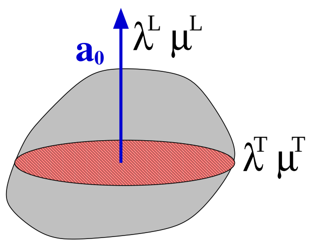


FIG. 2.15 – *Définition de coefficients de Lamé dans la direction privilégiée et dans le plan transverse*

Considérons que z soit la direction privilégiée, c'est-à-dire prenons $\mathbf{a}_0 = (0 \ 0 \ 1)$. On doit alors modifier tous les termes de l'énergie faisant intervenir la direction z. On peut dès maintenant séparer ces termes en deux groupes, d'une part les termes liés à l'étirement et d'autre part ceux liés au cisaillement. On verra un peu plus loin que ces deux types d'anisotropie interviennent bien spécifiquement dans la modification du comportement biomécanique.

Nous allons donc commencer par les termes liés à l'**étirement anisotrope** : ${\epsilon_z^l}^2$ et $\left(\epsilon_x^l \epsilon_z^l + \epsilon_y^l \epsilon_z^l\right)$. Pour modifier leurs contributions à l'énergie de déformation, on définit deux termes correctifs :

$$\Delta W^l_{\text{Étir\_aniso1}} = \left(\frac{\Delta\lambda}{2} + \Delta\mu\right) {\epsilon_z^l}^2 \tag{2.37}$$

$$\Delta W^l_{\text{Étir\_aniso2}} = \Delta\lambda \left(\epsilon_x^l \epsilon_z^l + \epsilon_y^l \epsilon_z^l\right) \tag{2.38}$$

Il est maintenant intéressant de relier ces termes d'énergie au tenseur de déformation E_l et à la direction privilégiée $\mathbf{a}_0$. Pour cela, on définit un nouveau terme que l'on considérera comme un nouvel invariant :

$$I_4 = \mathbf{a}_0^t \ E_l \ \mathbf{a}_0 \tag{2.39}$$

On montre alors que les termes d'énergie précédents s'expriment en fonction de I_4 et d'un autre terme intervenant dans l'énergie élastique isotrope, $I_1 = trE_l$:

$$\Delta W^l_{\text{Étir\_aniso1}} = \left(\frac{\Delta\lambda}{2} + \Delta\mu\right) \left(\mathbf{a}_0^t \ E_l \ \mathbf{a}_0\right)^2 \qquad = \left(\frac{\Delta\lambda}{2} + \Delta\mu\right) {I_4}^2 \tag{2.40}$$

$$\Delta W^l_{\text{Étir\_aniso2}} = \Delta\lambda \Big[trE_l \ \left(\mathbf{a}_0^t \ E_l \ \mathbf{a}_0\right) - \left(\mathbf{a}_0^t \ E_l \ \mathbf{a}_0\right)^2\Big] \qquad = \Delta\lambda \left(I_1 \ I_4 - {I_4}^2\right) \tag{2.41}$$

De la même manière, on doit corriger la pondération des termes liés au **cisaillement anisotrope**. Pour cela on définit un autre terme d'énergie qui est égal à :

$$\Delta W^l_{\text{cisail\_aniso}} = \frac{\Delta\mu}{2} \left({\gamma_{xz}^l}^2 + {\gamma_{yz}^l}^2\right) \tag{2.42}$$

Comme dans le cas de l'étirement, on est amené à introduire un dernier invariant, qui va nous permettre d'exprimer de façon générale le terme de cisaillement anisotrope en fonction du tenseur de déformation et de la direction d'anisotropie :

$$I_5 = \mathbf{a}_0^t \ {E_l}^2 \ \mathbf{a}_0 \tag{2.43}$$

$$\Delta W^l_{\text{cisail\_aniso}} = \frac{\Delta\mu}{2} \Big[\mathbf{a}_0^t \ {E_l}^2 \ \mathbf{a}_0 - 4\left(\mathbf{a}_0^t \ E_l \ \mathbf{a}_0\right)^2\Big] = \frac{\Delta\mu}{2} \left(I_5 - 4{I_4}^2\right) \tag{2.44}$$

Nous venons donc de définir de nouveaux termes qui, ajoutés à la loi de *Hooke*, permettent de prendre en compte la présence d'une direction privilégiée dans le matériau. Cette contribution anisotrope à l'énergie de déformation élastique se décompose en deux termes d'étirement et un terme de cisaillement. On peut donc choisir d'utiliser les termes que l'on veut, éventuellement en les pondérant les uns par rapport aux autres, suivant le comportement que l'on veut modéliser.

Si l'on veut tenir compte à la fois de l'étirement et du cisaillement anisotrope, on utilise le terme correctif suivant :

$$\Delta W^l_{\text{Aniso}} = \Delta\lambda\ I_1 I_4 + 2\Delta\mu\ I_5 - \Big(\frac{\Delta\lambda}{2} + \Delta\mu\Big) {I_4}^2, \tag{2.45}$$

ce qui permet d'obtenir l'énergie de déformation pour un matériau élastique linéaire transversalement isotrope, qui s'écrit sous la forme :

$$\begin{aligned} W^l_{\text{Trans\_iso}} =& \Delta W^l + \Delta W^l_{\text{Aniso}} \\ =& \boxed{\frac{\lambda}{2}\ {I_1}^2 + \mu\ I_2} + \boxed{\Delta\lambda\ I_1 I_4 + 2\Delta\mu\ I_5 - \Big(\frac{\Delta\lambda}{2} + \Delta\mu\Big) {I_4}^2} \\ =& \boxed{\frac{\lambda}{2}\ \big(trE_l\big)^2 + \mu\ tr\ {E_l}^2} \\ +& \boxed{\Delta\lambda\ \big(trE_l\big)\big(\mathbf{a}_0^t\ E_l\ \mathbf{a}_0\big) + 2\Delta\mu\ \big(\mathbf{a}_0^t\ {E_l}^2\ \mathbf{a}_0\big) - \Big(\frac{\Delta\lambda}{2} + \Delta\mu\Big)\big(\mathbf{a}_0^t\ E_l\ \mathbf{a}_0\big)^2} \end{aligned} \tag{2.46}$$

Finalement, on peut dériver cette énergie par rapport à E pour obtenir le second tenseur de contraintes de Piola-Kirchhoff, qui est donné dans le cas isotrope linéaire par l'équation 1.28.

$$\begin{aligned} \Sigma^l_{\text{Trans\_iso}} =& \frac{\partial W^l_{\text{Trans\_iso}}}{\partial E_l} \\ =& \lambda\ (trE_l)\ I + 2\mu\ E_l + \Delta\lambda\ \Big[\big(\mathbf{a}_0 \otimes \mathbf{a}_0\big)\ trE_l + \big(\mathbf{a}_0^t\ E_l\ \mathbf{a}_0\big)\ I\Big] \\ +& 2\Delta\mu\ \Big[\big(\mathbf{a}_0 \otimes \mathbf{a}_0\big)\ E_l + E_l\ \big(\mathbf{a}_0 \otimes \mathbf{a}_0\big)\Big] - \big(\Delta\lambda + 2\Delta\mu\big)\ \big(\mathbf{a}_0^t\ E_l\ \mathbf{a}_0\big)\ \big(\mathbf{a}_0 \otimes \mathbf{a}_0\big) \end{aligned} \tag{2.47}$$

2.5.1.2 Formulation discrète par les éléments finis

Comme dans le cas de isotrope, on utilise la méthode des éléments finis pour discrétiser le problème. On va donc exprimer les termes liés à l'anisotropie du matériau à l'intérieur de chaque tétraèdre du maillage. Pour cela, il faut attribuer à chaque tétraèdre de nouvelles données, à savoir une direction d'anisotropie $\mathbf{a}_0$ et les coefficients de Lamé correspondant à cette direction privilégiée λ^L et μ^L. On rappelle que le tenseur de déformation E_l s'exprime à l'intérieur du tétraèdre $\mathcal{T}$ en fonction des déplacements de ses sommets et de ses fonctions de forme par :

$$E_l = \frac{1}{2}\sum_{j=0}^{3}\big(\mathbf{U}_j \otimes \boldsymbol{\alpha}_j + \boldsymbol{\alpha}_j \otimes \mathbf{U}_j\big), \tag{2.48}$$

ce qui nous permet de calculer les différents termes exprimant l'anisotropie du matériau, à savoir ${I_4}^2$, $I_1 I_4$ et I_5 :

$$\begin{aligned} {l_4}^2 &= \big({\mathbf{a}_0}^t\ E_l\ \mathbf{a}_0\big)^2 = \sum_{k,l} \mathbf{U}_k^t \Big[\big(\mathbf{a}_0 \otimes \mathbf{a}_0\big)\big(\boldsymbol{\alpha}_k \otimes \boldsymbol{\alpha}_l\big)\big(\mathbf{a}_0 \otimes \mathbf{a}_0\big)\Big]\mathbf{U}_l \\ l_1 l_4 &= tr\ E_l({\mathbf{a}_0}^t\ E_l\ \mathbf{a}_0) = \sum_{k,l} \mathbf{U}_k^t \Big[\big(\boldsymbol{\alpha}_k \otimes \boldsymbol{\alpha}_l\big)\big(\mathbf{a}_0 \otimes \mathbf{a}_0\big)\Big]\mathbf{U}_l \\ l_5 &= {\mathbf{a}_0}^t\ {E_l}^2\ \mathbf{a}_0 = \sum_{k,l} \mathbf{U}_k^t \Big[\big(\mathbf{a}_0 \otimes \mathbf{a}_0\big)\big(\boldsymbol{\alpha}_l \otimes \boldsymbol{\alpha}_k\big) + \big(\boldsymbol{\alpha}_l \otimes \boldsymbol{\alpha}_k\big)\big(\mathbf{a}_0 \otimes \mathbf{a}_0\big) \\ &\qquad + \big(\boldsymbol{\alpha}_l.\boldsymbol{\alpha}_k\big)\big(\mathbf{a}_0 \otimes \mathbf{a}_0\big) + \big(\mathbf{a}_0 \otimes \mathbf{a}_0\big) : \big(\boldsymbol{\alpha}_l.\boldsymbol{\alpha}_k\big)\ I\Big]\mathbf{U}_l \end{aligned} \tag{2.49}$$

où ":" est le produit scalaire de deux matrices (il est défini dans la partie 1.4). Pour plus de détails sur les calculs, voir annexe B.1.

Ainsi, la contribution anisotrope à l'énergie de déformation définie par l'équation 2.45 s'écrit à l'intérieur du tétraèdre $\mathcal{T}$:

$$\Delta W_{\text{Aniso}} = \sum_{j,k=0}^{3} \mathbf{U}_j^t [\mathcal{A}_{\mathcal{T}}^{jk}] \mathbf{U}_k, \tag{2.50}$$

$$\begin{aligned} \mathcal{A}_{\mathcal{T}}^{jk} &= \Delta\lambda(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k)(\mathbf{a}_0 \otimes \mathbf{a}_0) - \Big(\frac{\Delta\lambda}{2} + \Delta\mu\Big)(\mathbf{a}_0 \otimes \mathbf{a}_0)(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k)(\mathbf{a}_0 \otimes \mathbf{a}_0) \\ &+ \frac{\Delta\mu}{2}\Big[(\mathbf{a}_0 \otimes \mathbf{a}_0)(\boldsymbol{\alpha}_k \otimes \boldsymbol{\alpha}_j) + (\boldsymbol{\alpha}_k \otimes \boldsymbol{\alpha}_j)(\mathbf{a}_0 \otimes \mathbf{a}_0) + (\boldsymbol{\alpha}_j \,.\, \boldsymbol{\alpha}_k)(\mathbf{a}_0 \otimes \mathbf{a}_0) \\ &\qquad + (\mathbf{a}_0 \otimes \mathbf{a}_0) : (\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k) I\Big] \end{aligned} \tag{2.51}$$

De la même manière que dans le cas isotrope, $\mathcal{A}_{\mathcal{T}}^{jk}$ est le tenseur de rigidité anisotrope de l'arête $(\mathbf{P}_j,\mathbf{P}_k)$ (ou du sommet $\mathbf{P}_j$ si $j = k$). Cette énergie vient s'ajouter à la partie isotrope (équation 2.9) simplement en sommant les tenseurs de rigidité correspondants. La dérivation des forces exercées par le tétraèdre sur chacun de ses sommets (équation 2.11), ainsi que l'accumulation des tenseurs sur les sommets et les arêtes du maillage (équation 2.12), se font de manière parfaitement similaire.

On calcule finalement la force élastique exercée sur le sommet $\mathbf{P}_p$, pour un matériau linéaire transversalement isotrope, par la formule :

$$\boxed{\mathbf{F}_p^{\text{Trans\_iso}} = \big[\mathcal{G}^{pp}\big]\,\mathbf{U}_p + \sum_{\mathcal{A} \in \nu(\mathbf{P}_p)} \big[\mathcal{G}^{pj}\big]\,\mathbf{U}_j \quad \text{où} \quad \mathcal{G}^{pj} = \mathcal{B}^{pj} + \mathcal{A}^{pj}} \tag{2.52}$$

2.5.1.3 Résultats

La généralisation de l'élasticité linéaire aux matériaux transversalement isotropes se traduit donc simplement par une modification des tenseurs de contraintes, qui intègrent à présent une contribution anisotrope. Même si cette contribution entraîne un surcoût de calculs, cela n'aura aucune influence sur l'efficacité des modèles déformables qui en découlent. En effet, que ce soit avec le modèle pré-calculé (paragraphe 2.4.1) ou le modèle masse-tenseur (paragraphe 2.4.2), les tenseurs de rigidités sont calculés une fois pour toutes lors de la création du maillage. Par contre, l'utilisation de l'élasticité transversalement isotrope pourra entraîner certains problèmes de stabilité, dus aux différences de raideurs en fonction de la direction dans laquelle se fait la déformation. Dans le cas du modèle pré-calculé, cela se traduit par une matrice de rigidité globale encore plus mal conditionnée, ce qui entraîne une augmentation du nombre d'itérations de l'algorithme de gradient conjugué pour parvenir à une solution d'équilibre. Mais cela n'affectera que les pré-calculs. Pour le modèle masse-tenseur, cette instabilité se traduit par une diminution du pas temps, et donc par un temps de convergence plus long. Mais on peut limiter cette perte d'efficacité en utilisant un amortissement qui tienne compte de la rigidité locale du matériau dans les trois directions de l'espace (cf. paragraphe 2.4.2.3).

En ce qui concerne l'utilisation des deux modèles déformables, il n'y a aucune différence avec le modèle isotrope. Tout ce qui a été dit aux paragraphes 2.4.1 et 2.4.2 reste donc vrai.

Nous présentons dans les paragraphes suivants des exemples mettant en œuvre l'élasticité transversalement isotrope. Le premier exemple concerne la modélisation de structures de type tubulaire, qui peuvent représenter des vaisseaux sanguins, des tendons ou des muscles. Il permettra de mettre en évidence le caractère anisotrope des déformations obtenues, et de comparer ces déformations suivant que l'on utilise ou pas le terme de cisaillement anisotrope. Dans le second exemple, nous présentons la modélisation de matériaux hybrides possédant des parties isotropes et des parties anisotropes. De telles matériaux se rencontrent dans les tissus traversés par des fibres, ou dans certains organes dans lesquels l'arborescence vasculaire est si importante qu'elle intervient dans la façon dont l'organe se déforme. C'est notamment le cas du foie.

2.5.1.4 Modélisation de structures tubulaires

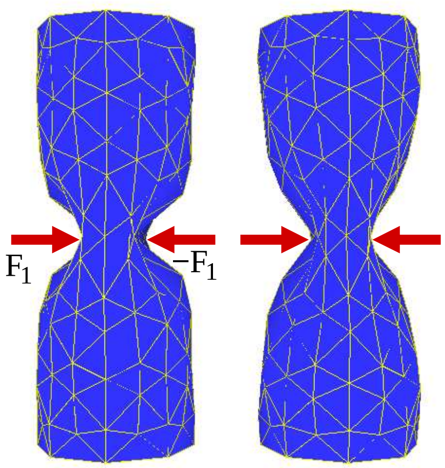
Dans ce premier exemple, nous considérons deux cylindres identiques. Le premier est isotrope (à gauche sur les figures 2.16(a) et 2.16(b)), alors que l'autre utilise l'élasticité transversalement isotrope, ce qui le rend 5 fois plus raide le long de sa hauteur (à droite sur les figures 2.16(a) et 2.16(b)). Nous utilisons ici un modèle anisotrope à la fois en étirement et en cisaillement.

Dans un premier temps, nous pinçons les deux cylindres en leur appliquant des forces perpendiculaires à la direction d'anisotropie (figure 2.16(a)). On peut voir que, même si l'on n'obtient pas exactement le même résultat, l'amplitude de la déformation est identique dans les deux cas. De plus, la déformation semble plus réaliste sur le cylindre transversalement isotrope. On verra d'ailleurs que c'est le terme de cisaillement anisotrope qui lisse la déformation le long de la direction privilégiée.

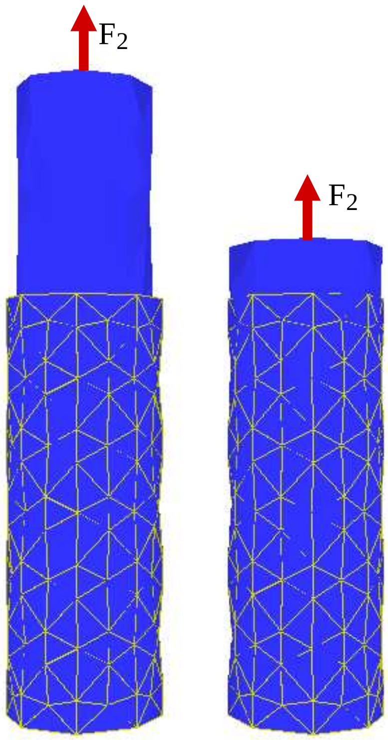
On applique ensuite sur la face supérieure des deux cylindres de la figure 2.16(b) la même force dirigée le long de la direction d'anisotropie (leurs faces inférieures étant fixées). Comme on l'espérait, le cylindre de droite, qui est plus raide dans cette direction, se déforme nettement moins.

Pour mettre en évidence le rôle que joue le terme de cisaillement anisotrope, nous avons renouvelé l'expérience, mais cette fois en n'utilisant que le terme d'étirement anisotrope. Si cette opération ne change strictement rien lorsque l'on étire le cylindre le long de sa direction d'anisotropie, elle apporte par contre un changement lorsque l'on pince le modèle dans le plan d'isotropie. Alors qu'avec le terme de cisaillement on obtenait une déformation sensiblement différente du modèle isotrope, cette fois les deux déformations sont quasiment parfaitement superposables (figure 2.17(a), (b) et (c)).

C'est donc le terme de cisaillement anisotrope qui conduit au lissage de la déformation le long de la direction privilégiée, comme on peut le voir sur la figure 2.17(d). Pour mieux comprendre ce phénomène, raisonons sur l'exemple en deux dimensions de la figure 2.18. Sans le terme pénalisant le cisaillement, on peut obtenir de fortes variations d'angles entre les arêtes des éléments (figure 2.18(b)). Par contre, si on rajoute le terme d'anisotropie en cisaillement, les éléments de volumes n'admettront que peu de variation des angles entre leurs arêtes, et donc pour arriver à la même déformation, il faudra que le cisaillement soit réparti sur plusieurs éléments le long de la direction d'anisotropie (figure 2.18(b)).

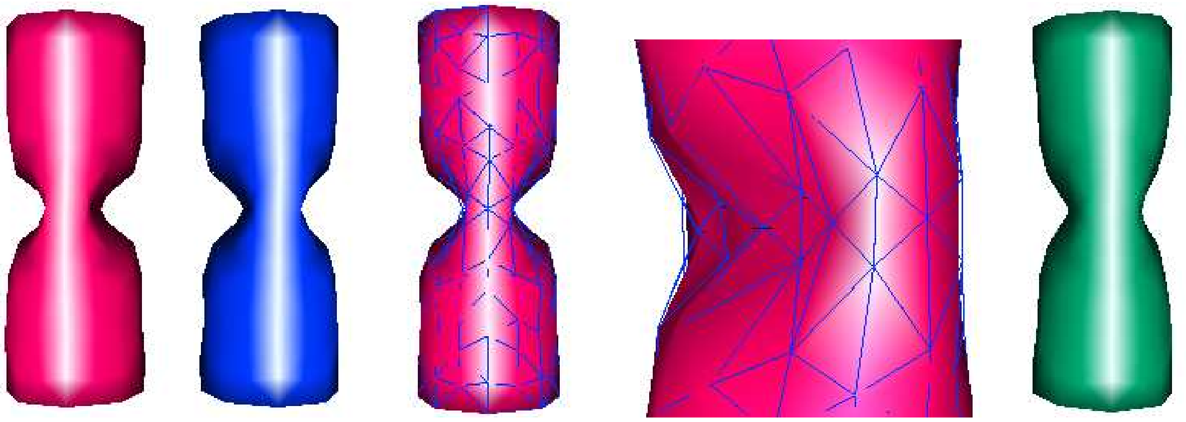


(a) Comparaison lorsque la déformation a lieu dans le plan d'isotropie : pincement des cylindres.



(b) Comparaison lorsque la déformation a lieu dans la direction d'anisotropie : étirement des cylindres.

FIG. 2.16 – *Comparaison entre un cylindre isotrope (cylindres de gauche des figures (a) et (b)) et un cylindre transversalement isotrope dont la direction privilégiée correspond à sa hauteur (cylindres de droite des figures (a) et (b)).*



(a) Comparaison du modèle isotrope (à gauche) et du modèle anisotrope en étirement (à droite) lorsqu'on leur applique les mêmes forces de pincement.

(b) Superposition des deux modèles de la figure (a).

(c) Gros plan de la figure (b). L'écart entre les deux déformations est vraiment minime.

(d) Modèle anisotrope en étirement et cisaillement.

FIG. 2.17 – *Comparaison entre des cylindres isotropes (gauche de la figure (a)), anisotropes en étirement (droite de la figure (a)), et complètement anisotropes (en étirement et en cisaillement), sur la figure (d).*

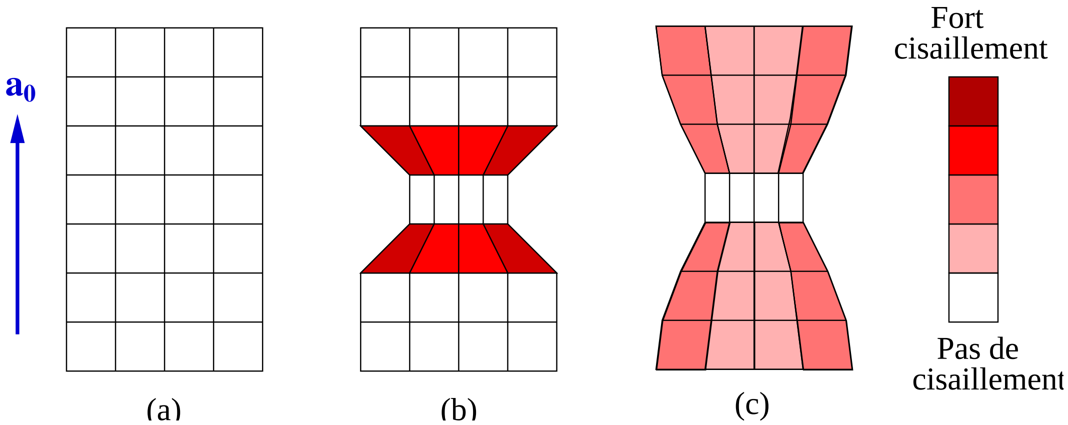


FIG. 2.18 – *Étude du cisaillement : variation d'angle entre les faces. (a) forme au repos. (b) sans le terme pénalisant le cisaillement, il peut être localement important. (c) avec le terme de cisaillement, les éléments de volume n'admettant que peu de cisaillement, il est réparti le long de la direction d'anisotropie.*

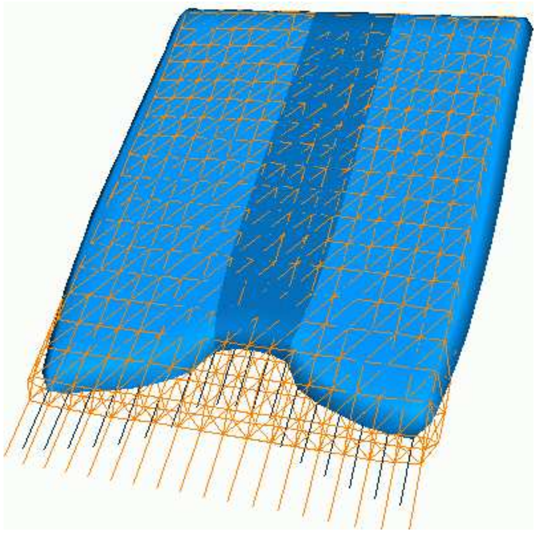
Ce premier exemple met donc bien en évidence l'intérêt que représente l'utilisation de l'élasticité transversalement isotrope pour modéliser des matériaux possédant une direction de déformation privilégiée. De plus, nous avons proposé une formulation dans laquelle nous séparons le terme d'étirement anisotrope du terme de cisaillement. Le premier donne au matériau une plus grande rigidité dans la direction d'anisotropie, et nous venons de montrer que le second a pour effet de lisser la déformation le long de la direction privilégiée. On a ainsi la possibilité de choisir les termes que l'on veut en fonction de comportement biomécanique recherché. D'après ce premier exemple, la modélisation de structures tubulaires de type vaisseaux ou tendons sera plus réaliste en utilisant le terme de cisaillement. Par contre lorsque c'est uniquement la plus grande raideur du matériau dans une direction donnée qui est importante, le terme d'anisotropie en étirement sera généralement suffisant. C'est le cas notamment dans le second exemple d'application que nous proposons maintenant.

2.5.1.5 Introduction de fibres anisotropes dans un matériau isotrope

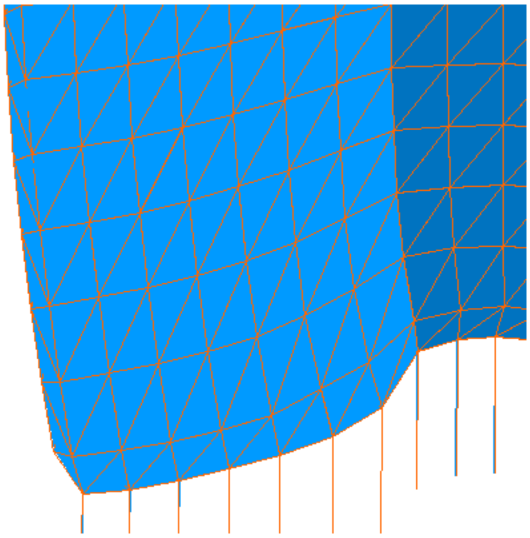
En plus de la modélisation de structures anatomiques anisotropes, nos modèles pré-calculé et masse-tenseur transversalement isotrope nous permettent d'inclure de telles structures dans des matériaux isotropes, créant ainsi une sorte de matériau hybride, dans lequel vont cohabiter des parties isotropes et des parties anisotropes. Ce genre de configuration se retrouve assez souvent dans les tissus vivants, en général à cause de la présence de fibres ou de vaisseaux sanguins au milieu d'un tissu isotrope.

Pour obtenir de tels modèles hybrides, il suffit de définir une zone du maillage sur laquelle on va ajouter la contribution anisotrope, le reste demeurant parfaitement isotrope. Dans l'exemple présenté sur la figure 2.19, nous avons défini une bande anisotrope représentée en foncé dans un matériau élastique isotrope. Nous avons choisi un modèle géométrique assez simple, afin d'avoir une meilleure compréhension de l'effet que va avoir cette partie anisotrope sur la déformation globale du modèle. La direction d'anisotropie correspond avec la plus grande dimension de la bande. Dans un premier temps nous avons étiré le modèle dans la direction d'anisotropie (figure 2.19(a)). On observe que la partie isotrope s'étire nettement plus que la bande anisotrope. La figure 2.19(b) montre l'important cisaillement local que subissent les éléments situés à la frontière entre la partie isotrope et la fibre. On a ensuite étiré le modèle dans la direction perpendiculaire à la direction de la fibre (figure 2.19(c)). On voit alors que, dans cette direction, la partie anisotrope ne perturbe pas la déformation, mais qu'elle se refuse à toute contraction le long de la direction d'anisotropie.

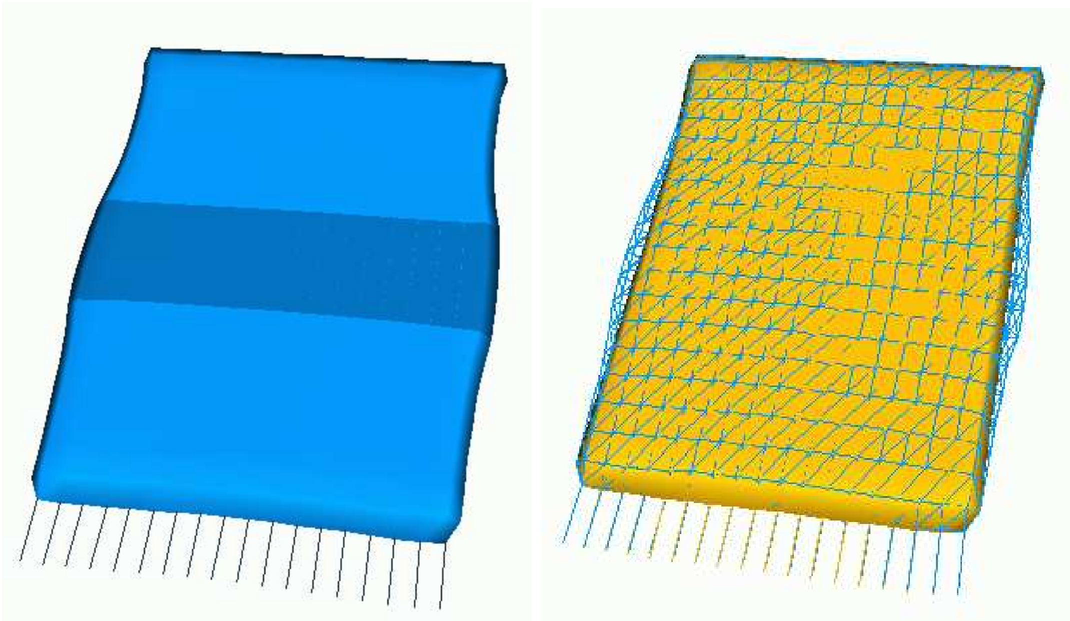
Finalement, nous avons utilisé ce modèle hybride afin de prendre en compte la présence de gros vaisseaux sanguins dans le foie. En effet le foie contient une grande quantité de vaisseaux sanguins, mais seulement les plus gros ont une réelle influence sur son comportement biomécanique. Nous avons donc construit un modèle synthétique d'arbre vasculaire à l'intérieur de notre modèle géométrique de foie. Cette arbre est composé de sept branches représentées par des segments. Pour chacune de ces lignes, nous déterminons les tétraèdres qu'elle traverse (figures 2.20(a)) et nous leur attribuons un comportement transversalement isotrope en prenant la direction de la ligne comme direction d'anisotropie (figure 2.20(c)). Nous obtenons ainsi un modèle déformable de foie qui tient



(a) Étirement dans la direction de la fibre. Le modèle contenant la fibre (en bleu) est comparé à un modèle isotrope (en jaune).



(b) Gros plan sur la frontière entre le matériau isotrope et la partie anisotrope. Cette zone est soumise à de fortes contraintes de cisaillement.



(c) Étirement du modèle hybride dans la direction orthogonale à la direction de la fibre. Il est ensuite comparé au même étirement sur le modèle isotrope.

FIG. 2.19 – *Effets de l'introduction d'une fibre anisotrope dans un matériau isotrope. Le matériau bleu est isotrope et la partie foncée représente une fibre dont la direction d'anisotropie correspond à sa plus grande dimension.*

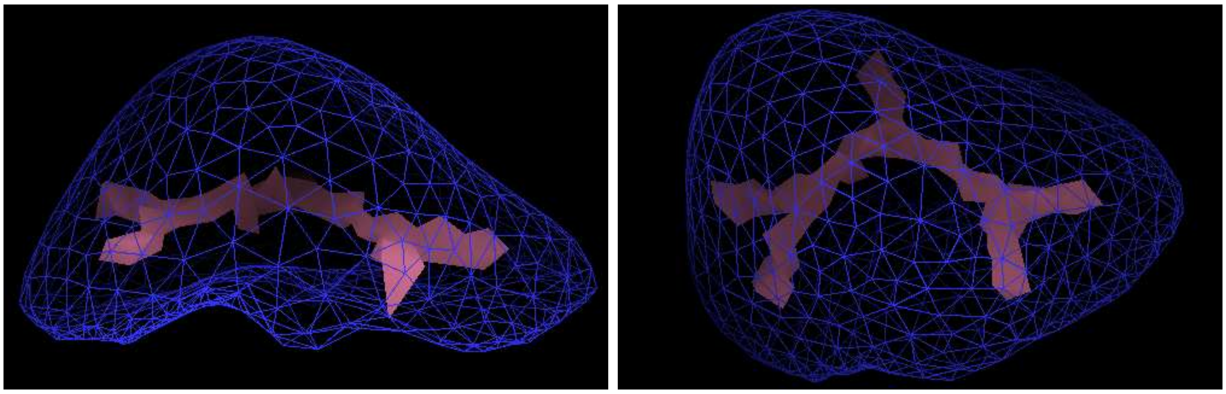
compte des propriétés anisotropes des vaisseaux sanguins qu'il contient (figures 2.20(d), 2.20(e) et 2.20(f)).

2.5.2 Anisotropie de surface

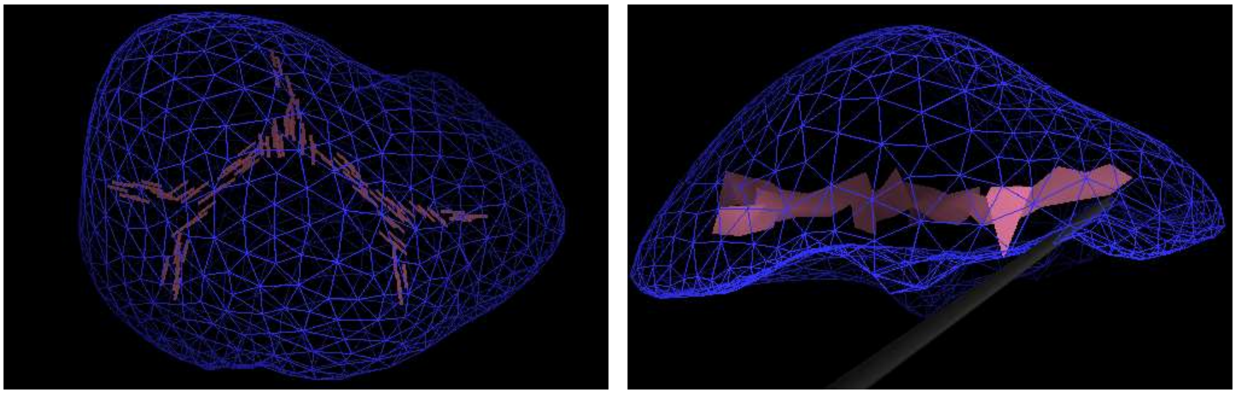
L'autre grande cause d'anisotropie parmi les structures anatomiques est la présence d'une membrane ou d'une peau à la surface des organes, venant renforcer leur structure. On a alors ce que nous appellerons une anisotropie de surface. C'est le cas en particulier pour le foie, qui possède à sa surface une peau élastique, la capsule de Glisson, qui est relativement rigide. Afin d'améliorer le réalisme biomécanique de notre modèle de foie, il semblait donc intéressant de tenir compte de cette propriété.

Il est possible, grâce à la théorie des membranes élastiques, de modéliser cette raideur surfacique en continuant dans la voie de l'élasticité et des éléments finis. C'est le cas du modèle de foie proposé par M. Vidrascu (Vidrascu, 1999), composé d'un matériau de Mooney-Rivlin incompressible complété d'une coque. Mais cette approche est très coûteuse en temps de calcul. Par contre, les réseaux masse-ressort se prêtent parfaitement à ce genre de comportement. Il suffira en effet d'augmenter la raideur des ressorts situés sur la surface de l'organe, voire même de définir un nouveau comportement (non-linéaire par exemple) pour ces ressorts. C'est l'approche choisie par D. d'Aulignac pour la cuisse (d'Aulignac et al., 1999a) et F. Boux de Casson pour le foie (Boux de Casson et Laugier, 1999).

Notre but étant, dans un premier temps, de ne pas pénaliser la fréquence de simulation, nous nous sommes tournés vers la solution d'un réseau masse-ressort, dont le fonctionnement est très proche de celui de notre modèle masse-tenseur. Nous modélisons ainsi la capsule de Glisson en rajoutant sur chaque arête de la surface du maillage un ressort de raideur k et de longueur au repos l_0. A chaque pas de temps, il faudra donc, en plus du calcul des forces dues au modèle masse-tenseur, parcourir les arêtes de la surface et calculer les forces que chaque ressort va exercer sur les deux sommets $\mathbf{P}_j$ et $\mathbf{P}_k$ auxquels il est relié. Cette force dérive de l'énergie élastique d'un ressort, et dépend de la raideur du ressort et de la variation de sa longueur :

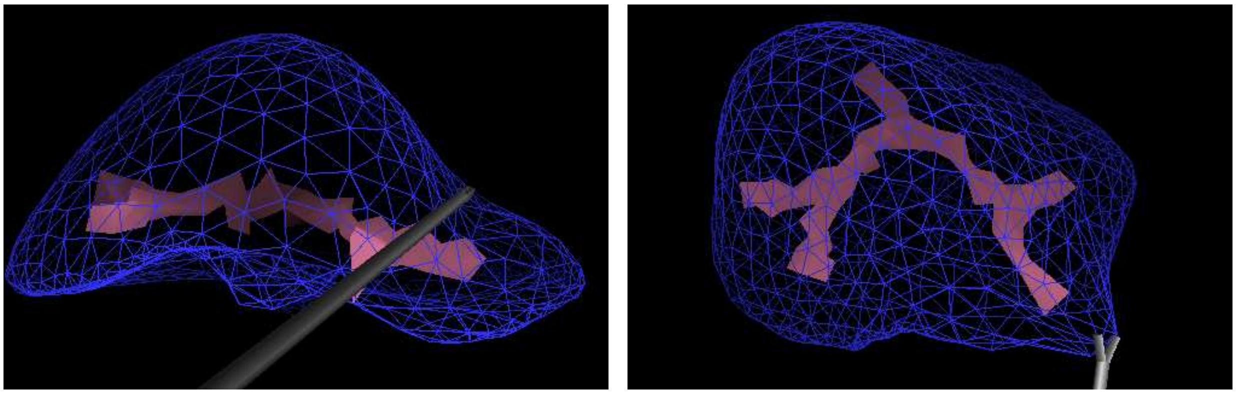


(a) Les tétraèdres représentés en plein contiennent les vaisseaux et ont un comportement élastique transversalement isotrope.



(b) Visualisation de la direction d'anisotropie dans les tétraèdres représentant les vaisseaux.

(c)



(d)

(e)

FIG. 2.20 – *Modèle déformable de foie hybride. Le parenchyme isotrope est traversé par des vaisseaux au comportement anisotrope.*

$$W^{\text{ressort}} = \frac{1}{2}k(l - l_0)^2, \qquad (2.53)$$

$$\begin{aligned} \mathbf{F}_j^{\text{ressort}} &= -\mathbf{F}_k^{\text{ressort}} \\ &= k\left(\frac{l - l_0}{l}\right)\overrightarrow{\mathbf{P}_j\mathbf{P}_k}. \end{aligned} \qquad (2.54)$$

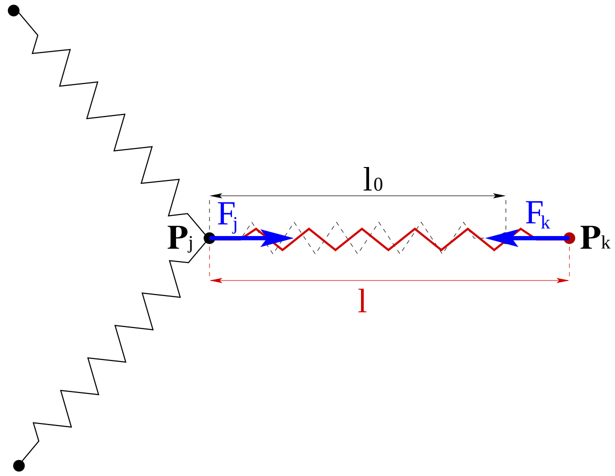


FIG. 2.21 – *Calcul des forces que chaque ressort exerce sur ces deux sommets.*

Le but de ce réseau masse-ressort surfacique étant de modéliser la présence de la capsule de Glisson, qui est relativement raide (voir paragraphe 5.3.1), nous devons choisir une raideur élevée des ressorts. Mais cela risque de provoquer des oscillations. C'est pourquoi nous considérons en fait un système masse-ressort amorti : chaque ressort est monté en parallèle avec un amortisseur visqueux qui pénalise la vitesse à laquelle varie la longueur du ressort. La force appliquée sur chaque extrémité du ressort vaut alors :

$$\mathbf{F}_j^{\text{ressort}} = -\mathbf{F}_k^{\text{ressort}} = k\left[(l - l_0) - \gamma\,\frac{\partial l}{\partial t}\right]\frac{\overrightarrow{\mathbf{P}_j\mathbf{P}_k}}{l} \qquad (2.55)$$

où γ est le cœfficient de pondération entre la raideur et l'amortissement.

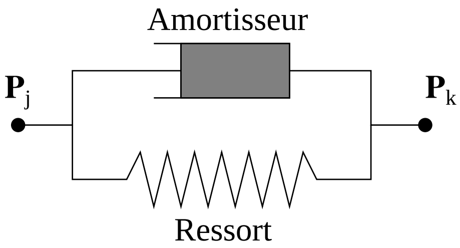


FIG. 2.22 – *Ressort couplé avec un amortisseur*

Un des points délicats lors de la mise en œuvre d'un modèle masse-ressort est le choix des raideurs des ressorts, car elles vont déterminer le caractère homogène de la déformation. La solution la plus simple consiste à prendre la même valeur partout. Mais on obtient alors un matériau qui sera plus rigide dans les zones où le maillage est plus fin (c'est-à-dire là où il y a plus de ressorts). Afin de contourner ce problème, Gelder et Wilhelms ont cherché à optimiser les valeurs des raideurs des ressorts de manière à ce que le comportement du matériau obtenu se rapproche le plus possible de celui d'une membrane élastique exprimée au travers de la méthode des éléments finis (Gelder et Wilhelms, 1997; Gelder, 1998). Pour cela, ils ont identifié les déformations obtenues en modélisant un triangle à l'aide de ressorts à celles obtenues en le considérant comme une membrane élastique.

Ils ont alors abouti à une raideur de chaque ressort du triangle valant :

$$k_c = \left(\frac{E}{1+\nu}\right)\frac{2\mathcal{A}}{c^2} + \left(\frac{E\,\nu}{1-\nu^2}\right)\frac{a^2+b^2-c^2}{4\mathcal{A}} \tag{2.56}$$

où a, b et c sont les longueurs des arêtes, $\mathcal{A}$ est l'aire du triangle et E et ν sont le module de Young et le rapport de Poisson du matériau simulé. Cette formule donne la contribution d'un triangle. Or, dans un maillage surfacique, chaque arête (et donc chaque ressort) appartient à deux triangles. On obtient donc la raideur de chaque ressort en sommant les contributions de ses deux triangles. Les auteurs ont appliqué ce principe pour représenter la peau d'un modèle de singe animé relativement élaboré (Wilhelms et VanGelder, 1997; Wilhelms et Gelder, 1997). Ils ont d'autre part montré que le fait de prendre $\nu = 0$ ne changait pratiquement rien à la déformation. Nous avons donc utilisé le même principe pour calculer les raideurs des ressorts de notre membrane :

$$k = E\,\frac{2(\mathcal{A}_1+\mathcal{A}_2)}{l^2}$$

$$\mathcal{A}_1 = \frac{1}{2}h_1\,l \quad \text{et} \quad \mathcal{A}_2 = \frac{1}{2}h_2\,l$$

$$\mathbf{F}_j^{ressort} = k\,\Delta l\,\vec{T}_{jk} = E\,\frac{h}{l}\,\Delta l\,\vec{T}_{jk} \tag{2.57}$$

où $\vec{T}_{jk}$ est le vecteur unitaire tangent au ressort dans sa position déformée.

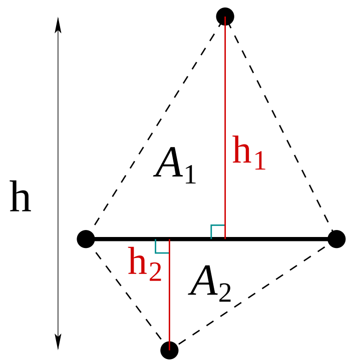
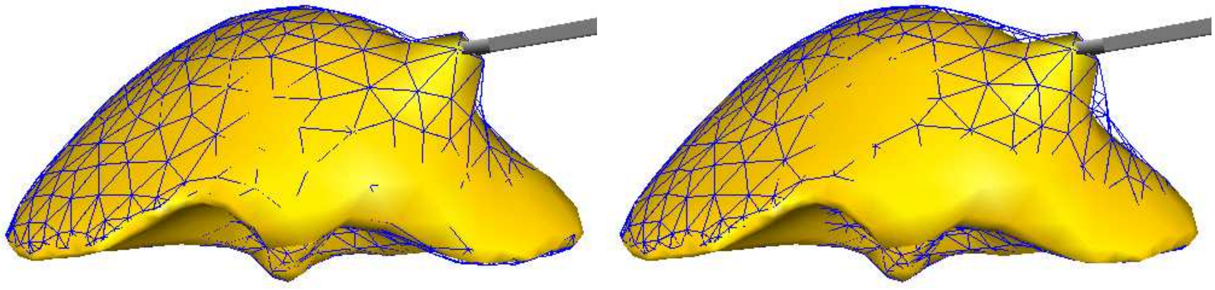
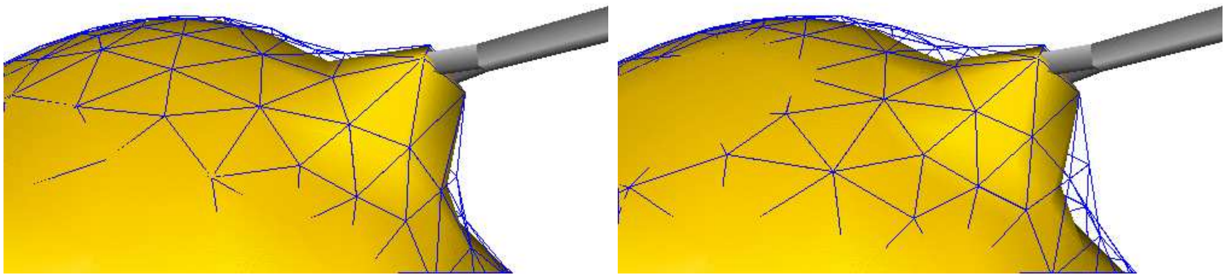


FIG. 2.23 – *Calcul de la raideur du ressort en fonction des aires des deux triangles adjacents.*

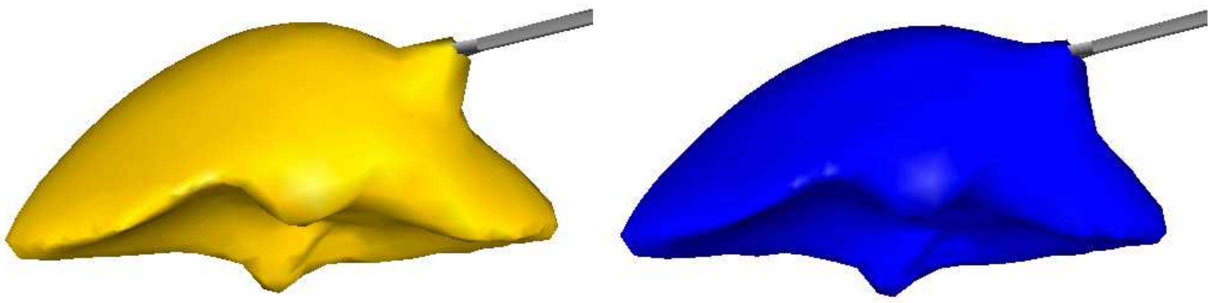
Nous avons rajouté cette membrane externe masse-ressort sur la surface de notre modèle masse-tenseur. Les calculs supplémentaires étant très simples et ne concernant que les arêtes de la surface du maillage, le temps de calcul est presque inchangé (on ne constate qu'une diminution de la fréquence de simulation d'environ 4 %). La figure 2.24 compare les comportements d'un modèle masse-tenseur et du même modèle sur la surface duquel nous avons rajouté un réseau masse-ressort afin de simuler la raideur surfacique provoquée par la capsule de Glisson. Les deux figures 2.24(a) et (b) montrent l'influence de la raideur de cette membrane sur les déformations globales et locales de notre modèle de foie. Entre le modèle de gauche et celui de droite la raideur des ressorts de la surface a été multipliée par 10. La figure 2.24(c) compare les déformations globales sans et avec la Capsule de Glisson. Ces expériences montrent que la présence d'une membrane élastique sur la surface du modèle modifie son comportement, en particulier au niveau du point de contact avec les instruments. L'introduction de cette anisotropie de surface sur notre modèle masse-tenseur améliore son réalisme biomécanique puisque les déformations tiennent maintenant compte de la présence de la capsule de Glisson sur la surface du foie.



(a) Le modèle représenté en "fil de fer" possède une membrane externe masse-ressort, alors que l'autre est isotrope. Entre la figure de gauche et celle de droite, la raideur des ressorts de la membrane a été multipliée par 10.



(b) Gros plans sur les expériences présentées ci-dessus.



(c) Comparaison entre les déformations globales des deux modèles, à gauche le modèle masse-tenseur isotrope et à droite le même modèle avec la membrane externe.

FIG. 2.24 – *Comparaison des déformations d'un modèle isotrope et du même modèle avec une membrane externe élastique représentant la capsule de Glisson.*

2.6 Conclusion

Nous avons présenté, dans ce chapitre, deux modèles déformables construits sur les bases de l'élasticité linéaire et la méthode des éléments finis. Ces deux modèles permettent de simuler en temps-réel les déformations d'organes tels que le foie, chacun avec ses avantages et ses inconvénients. Le premier tire sa grande efficacité de pré-calculs qui lui interdisent tout changement de topologie. De son coté, le modèle masse-tenseur permet de simuler découpes et déchirements, mais est plus coûteux en temps de calcul.

Nous nous sommes ensuite attachés à améliorer le réalisme biomécanique de ces modèles en les enrichissant d'un comportement anisotrope. Nous avons tout d'abord généralisé l'élasticité linéaire aux matériaux transversalement isotropes, c'est-à-dire possédant une direction privilégiée de déformation, et nous avons montré comment appliquer ce modèle pour représenter des tissus contenant des fibres. Nous avons ensuite ajouté à notre modèle dynamique masse-tenseur, une membrane externe de type masse-ressort permettant de modéliser la peau relativement raide qui compose la surface du foie (Capsule de Glisson).

Toutes ces contributions nous permettent de simuler les déformations d'une grande variété de structures anatomiques en temps-réel, avec un bon réalisme à la fois biomécanique et visuel.

Chapitre 3

L'élasticité en grands déplacements

Nous proposons, dans ce chapitre, un nouveau modèle déformable, basé sur l'élasticité de St Venant-Kirchhoff et la méthode des éléments finis. Ce modèle reste valable pour les grands déplacements, ce qui veut dire en particulier qu'il est invariant par rotation. Cette propriété améliore le réalisme de la déformation et résout les problèmes liés aux limitations de l'élasticité linéaire, qui n'est valable que pour les petits déplacements. Nous proposons deux versions de ce modèle. La première est une généralisation du modèle masse-tenseur présenté dans le chapitre précédent et la seconde met en œuvre une formulation plus originale intégrant implicitement la propriété d'invariance par rotation.

3.1 Introduction

Dans les chapitres précédents, nous avons utilisé une loi de déformation élastique linéaire. Nous avons défini deux modèles déformables inspirés des travaux de Stéphane Cotin, le modèle pré-calculé et le modèle masse-tenseur. Nous nous sommes ensuite attachés à améliorer leur comportement biomécanique en y ajoutant la possibilité de modéliser des matériaux anisotropes. Cependant, même si ces modèles déformables nous permettent de simuler les déformations des structures anatomiques dont nous avons besoin pour notre prototype de simulateur de chirurgie, leur point faible s'avère être leur comportement élastique linéaire. Du fait de l'utilisation d'une relation linéaire entre le tenseur de déformation E_l et le gradient du champ de déplacements, les modèles ne sont plus invariants par rotation. Il en résulte des distortions très peu réalistes dès que l'hypothèse des *petits déplacements* n'est plus vérifiée. C'est pourquoi nous allons nous intéresser, dans ce paragraphe, à l'équivalent en *grands déplacements* du modèle d'élasticité linéaire, à savoir le modèle de St Venant-Kirchhoff.

L'élasticité non-linéaire est en général utilisée lorsque les principaux objectifs sont la précision et le réalisme biomécanique de la déformation, et non pas l'efficacité et le fonctionnement en temps réel. Par exemple M. Vidrascu utilise un modèle de Mooney-Rivlin incompressible pour simuler les déformations du foie (Vidrascu, 1999). Malgré l'utilisation d'une méthode de décomposition de domaine, les calculs prennent plusieurs minutes. Vuskovic et Kauer utilisent eux aussi des modèles non-linéaires du foie (loi néo-Hookéenne et modèle de Veronda-Westmann) pour calculer les constantes d'élasticité en identifiant les déformations simulées et des données mesurées *in-vivo* (Vuskovic et al., 1999b; Vuskovic et al., 1999a; Vuskovic et al., 2000; Kauer et al., 1999). Cette fois encore le temps réel n'est pas nécessaire. Des modèles d'élasticité non-linéaire sont utilisés pour simuler les déformations d'autres organes, comme par exemple le cerveau afin de prédire les déformations engendrées par le *brain shift* ou par l'apparition (ou l'ablation) d'une tumeur (Kyriacou et al., 1998; Kyriacou et al., 1999), ou encore la peau dans le but d'en étudier l'élasticité (Tsap et al., 1998). On trouve aussi des applications de l'élasticité de St Venant-Kirchhoff en animation (Hirota et al., 2000).
Les exemples d'utilisation de modèles non-linéaires dans des applications temps-réel sont rares. Pour la simulation de chirurgie gynécologique, Szekely *et al.* utilisent un modèle de Mooney-Rivlin (Székely et al., 1998; Székely et al., 2000b; Székely et al., 2000a) qui fonctionne en temps réel par le biais d'une parallélisation massive (Rhomberg et al., 1999). D'autre part, Zhuang utilise un modèle d'élasticité en grands déplacements (apparemment relativement proche de ce que nous allons présenter) pour simuler la manipulation d'objets dans un environnement virtuel (Zhuang et Canny, 1999; Zhuang et Canny, 2000; Zhuang, 2000). À notre connaissance, aucun de ces modèles ne se prête aux changements de topologie.

Dans ce chapitre, nous proposons un nouveau modèle déformable dynamique temps-réel qui s'appuie sur l'expérience acquise avec le modèle masse-tenseur, mais qui reste valable pour les grands déplacements.
Nous allons tout d'abord mettre en évidence les limitations du modèle d'élasticité linéaire et en particulier les problèmes que pose l'absence d'invariance par rotation. Le chapitre se décompo-

sera ensuite en deux parties principales. La première (paragraphe 3.3) traite de la généralisation du modèle masse-tenseur à l'élasticité non-linéaire de St Venant-Kirchhoff. Après en avoir détaillé la formulation, nous nous intéresserons aux problèmes de conservation de volume en y ajoutant des contraintes d'incompressibilité. L'élasticité non-linéaire entraînant un surcroît de calcul, nous proposerons ensuite un modèle adaptatif permettant d'optimiser les performances. Enfin, nous généraliserons son comportement aux matériaux transversalement isotropes.
La deuxième grande partie (paragraphe 3.4) introduit une formulation originale du même modèle qui intègre naturellement la notion d'invariance par rotation, et dont nous détaillerons la mise en œuvre dans les cas 2D et 3D.

3.2 Les limites de l'élasticité linéaire

Si l'élasticité linéaire possède de nombreux avantages en terme de simplicité des équations et de possibilités d'optimisation, elle présente néanmoins une forte restriction, puisqu'elle est limitée aux petits déplacements. On admet en général que cette loi d'élasticité n'est valable que pour des déplacements inférieurs à 10 % de la taille totale du domaine considéré (Fung, 1993; Maurel et al., 1998). En réalité, cela signifie simplement qu'il faut se restreindre à des déplacements suffisamment petits pour que les rotations engendrées par ces déplacements puissent être approximées par des translations. En effet, du fait de l'utilisation de la version linéarisée du tenseur de déformation, le modèle n'est plus invariant par rotation. Lorsqu'un élément de volume subit une rotation, il l'interprète comme une augmentation de l'énergie élastique, qui se traduit par une distortion. La figure 3.1 montre la déformation que subit un modèle élastique linéaire lorsqu'il est soumis à une rotation.

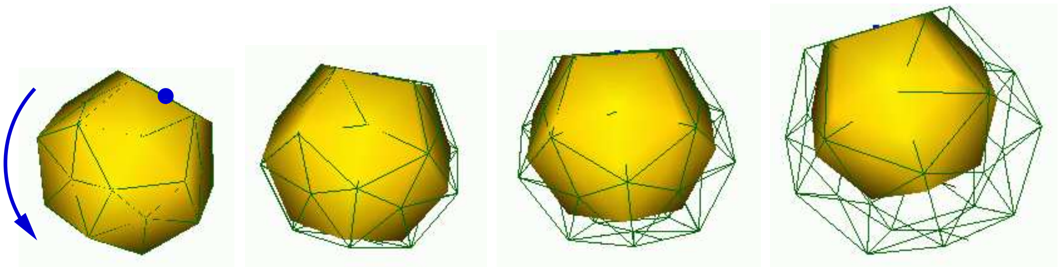
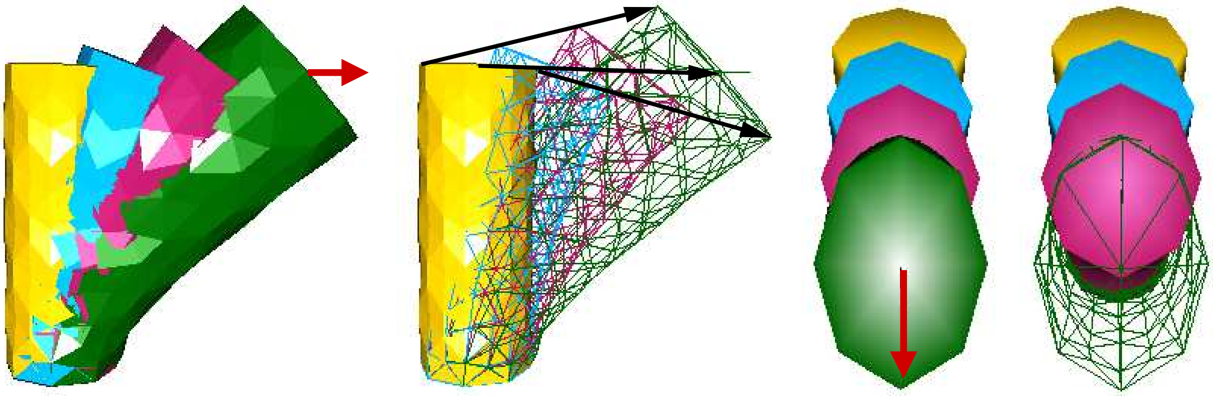


FIG. 3.1 – *Dilatation d'un matériau élastique linéaire (fil de fer) sous l'effet d'une rotation globale.*

Lorsqu'il s'agit d'une rotation globale, on pourrait imaginer résoudre le problème par un changement de repère approprié. Mais dans la plupart des cas, il n'y a qu'une partie de l'objet simulé qui subit une rotation par rapport au reste du domaine. C'est en particulier le cas dès lors qu'une partie de l'objet est fixée, et que le reste est soumis à un ensemble de contraintes entraînant une rotation locale. La figure 3.2 montre un cylindre dont la face inférieure a été fixée. La face supérieure est soumise à une force dirigée vers la droite. Nous avons représenté en (a) les déformations entraînées pour plusieurs valeurs successives de l'intensité de la force. On observe, comme sur la figure 3.1, une dilatation du modèle aux endroits où il subit une rotation. Sur la figure 3.2(b), nous avons représenté à l'aide de flèches les trajectoires de certains points sur la face supérieure du cylindre. Nous constatons que ces points suivent, au cours de la déformation, une trajectoire linéaire qui est responsable de la dilatation observée. Cette dilatation est d'autant plus irréaliste qu'elle n'est pas isotrope, puisqu'elle ne se produit que dans le plan de rotation. On le constate en observant la déformation subie par la face supérieure du cylindre (3.2(c)).



(a) Vue de coté.

(b) Vue de coté en rendu fil de fer, avec une représentation des trajectoires suivies par certains points.

(c) Vues du dessus.

FIG. 3.2 – *Plusieurs déformations successives d'un cylindre élastique linéaire, entraînant des rotations de plus en plus importantes de la partie supérieure du modèle.*

Les limites que montre le modèle d'élasticité linéaire lorsqu'il subit des grands déplacements nous ont amenés à considérer un modèle d'élasticité très proche du modèle linéaire mais qui reste valable pour les grandes déformations, le modèle d'élasticité de St Venant-Kirchhoff.

3.3 Le modèle d'élasticité de St Venant-Kirchhoff

Nous avons introduit le modèle d'élasticité de St Venant-Kirchhoff au paragraphe 1.2.6, avant de nous limiter dans le chapitre 2 au cas particulier de l'élasticité linéaire. Nous rappelons ici que ce modèle d'élasticité repose sur le tenseur de déformation de Green-St Venant E :

$$E = \frac{1}{2}\big(\nabla\mathbf{U}^t + \nabla\mathbf{U} + \nabla\mathbf{U}^t\nabla\mathbf{U}\big), \tag{3.1}$$

qui sert de base à l'expression de l'énergie de déformation :

$$W = \frac{\lambda}{2}\,\big(trE\big)^2 + \mu\; trE^2. \tag{3.2}$$

L'énergie de déformation, qui est une fonction quadratique du gradient du champ de déplacements dans le cas linéaire, est maintenant un polynôme de degré quatre par rapport à $\nabla \mathbf{U}$ (les détails des calculs se trouvent en annexe A.2) :

$$\begin{aligned} W = &\frac{\lambda}{2}\left[(div\ \mathbf{U}) + \frac{1}{2}\|\nabla\mathbf{U}\|^2\right]^2 + \mu\|\nabla\mathbf{U}\|^2 - \frac{\mu}{2}\|rot\ \mathbf{U}\|^2 \\ &+ \mu(\nabla\mathbf{U} : \nabla\mathbf{U}^t\nabla\mathbf{U}) + \frac{\mu}{4}\|\nabla\mathbf{U}^t\nabla\mathbf{U}\|^2, \end{aligned} \tag{3.3}$$

Ce modèle n'étant plus linéaire par rapport au champ de déplacements, nous ne pouvons plus utiliser la méthode des pré-calculs. Nous allons donc appliquer à ce modèle de St Venant-Kirchhoff un schéma de résolution de type masse-tenseur. Le paragraphe suivant s'attache à exprimer les équations que nous venons de présenter en termes d'éléments finis, pour en déduire un modèle déformable masse-tenseur non-linéaire.

3.3.1 Le modèle masse-tenseur non-linéaire

Comme dans le cas de l'élasticité linéaire, nous allons maintenant exprimer le tenseur de déformation, l'énergie et la force élastique au travers de la méthode des éléments finis. Afin de ne pas trop alourdir le document, nous ne donnerons ici que les résultats nécessaires à la bonne compréhension des nouveaux développements. Les lecteurs intéressés par les étapes de calcul, ou bien voulant mettre en œuvre la méthode, se reporteront à l'annexe B.2.

En conservant les conventions de notations utilisées pour l'élasticité linéaire, nous exprimons l'énergie de déformation d'un tétraèdre $\mathcal{T}$ pour le modèle d'élasticité de St Venant-Kirchhoff sous la forme :

$$\begin{aligned} W(\mathcal{T}) = &\sum_{j,k} \mathbf{U}_j^t \left[\mathcal{B}_{\mathcal{T}}^{jk}\right] \mathbf{U}_k + \sum_{j,k,l} \left(\mathbf{U}_j \,.\, \mathcal{C}_{\mathcal{T}}^{jkl}\right) (\mathbf{U}_k \,.\, \mathbf{U}_l) \\ &+ \sum_{j,k,l,m} \mathcal{D}_{\mathcal{T}}^{jklm} (\mathbf{U}_j \,.\, \mathbf{U}_k) (\mathbf{U}_l \,.\, \mathbf{U}_m). \end{aligned} \tag{3.4}$$

Comme on pouvait s'y attendre, cette énergie comporte maintenant, en plus du terme quadratique de l'élasticité linéaire, des termes d'ordre trois et quatre mettant en œuvre respectivement les déplacements de trois et quatre sommets du tétraèdre. Ces termes font apparaître les nouvelles grandeurs $\mathcal{C}_{\mathcal{T}}^{jkl}$ et $\mathcal{D}_{\mathcal{T}}^{jklm}$, qui sont respectivement des vecteurs et des scalaires de rigidité. Ils viennent compléter les tenseurs de rigidité $\mathcal{B}_{\mathcal{T}}^{jk}$ (équation 2.9) et dépendent des fonctions de forme du tétraèdre (équation 1.34) et des coefficients de Lamé du matériau :

$$\left\{ \begin{aligned} \mathcal{C}_{\mathcal{T}}^{jkl} &= \frac{\lambda}{2}\,\alpha_j\,(\alpha_k \,.\, \alpha_l) + \frac{\mu}{2}\left[\alpha_l\,(\alpha_j \,.\, \alpha_k) + \alpha_k\,(\alpha_j \,.\, \alpha_l)\right], && j,k,l = 0..3 \\ \mathcal{D}_{\mathcal{T}}^{jklm} &= \frac{\lambda}{8}\,(\alpha_j \,.\, \alpha_k)\,(\alpha_l \,.\, \alpha_m) + \frac{\mu}{4}(\alpha_j \,.\, \alpha_m)(\alpha_k \,.\, \alpha_l), && j,k,l,m = 0..3 \end{aligned} \right. \tag{3.5}$$

Afin de simplifier la formulation des équations, on peut montrer quelques propriétés de symétrie des différentes informations de rigidité :

$$\begin{array}{rcl} \mathcal{B}_{\mathcal{T}}^{jk} & = & \left(\mathcal{B}_{\mathcal{T}}^{kj}\right)^t \\ \mathcal{C}_{\mathcal{T}}^{jkl} & = & \mathcal{C}_{\mathcal{T}}^{jlk} \\ \mathcal{D}_{\mathcal{T}}^{jklm} & = & \mathcal{D}_{\mathcal{T}}^{kjml} = \mathcal{D}_{\mathcal{T}}^{lmjk} \end{array} \tag{3.6}$$

La force exercée par le tétraèdre sur chacun de ses sommets $\mathbf{P}_p$ est alors dérivée de l'énergie élastique $W(\mathcal{T})$ pour donner une équation polynomiale de degré trois :

$$\begin{aligned} \mathbf{F}^p(\mathcal{T}) = & \underbrace{2\sum_j \left[\mathcal{B}_{\mathcal{T}}^{pj}\right]\mathbf{U}_j}_{\mathbf{F}_1^p(\mathcal{T})} + \underbrace{\sum_{j,k} 2\,(\mathbf{U}_k \otimes \mathbf{U}_j)\,\mathcal{C}_{\mathcal{T}}^{jkp} + (\mathbf{U}_j.\mathbf{U}_k)\,\mathcal{C}_{\mathcal{T}}^{pjk}}_{\mathbf{F}_2^p(\mathcal{T})} \\ & + \underbrace{4\sum_{j,k,l} \mathcal{D}_{\mathcal{T}}^{jklp}\,\mathbf{U}_l\,\mathbf{U}_k^t\,\mathbf{U}_j}_{\mathbf{F}_3^p(\mathcal{T})} \;. \end{aligned} \tag{3.7}$$

On remarque que le premier terme de la force élastique, $\mathbf{F}_1^p(\mathcal{T})$, correspond exactement à la force rencontrée dans le cas de l'élasticité linéaire.

Cette expression de la force est valable pour l'élément de volume que représente le tétraèdre $\mathcal{T}$. Il faut maintenant accumuler les contributions de chaque tétraèdre, afin d'obtenir la force globale exercée sur un sommet du maillage. Dans le cas du modèle masse-tenseur linéaire, nous avions décomposé la force $\mathbf{F}_1^p(\mathcal{T})$ en deux, une partie étant reliée au déplacement du sommet lui-même et une seconde partie aux déplacements de ses voisins (équation 2.11) :

$$\mathbf{F}_1^p(\mathcal{T}) = \mathbf{F}^p_{\text{sommet p}}(\mathbf{U}_p) + \sum_{j\neq p} \mathbf{F}^p_{\text{aréte (p,j)}}(\mathbf{U}_j) \tag{3.8}$$

Nous avions alors additionné les tenseurs de rigidité correspondants sur les sommets et les arêtes du maillage (équation 2.12).

Le même principe peut être appliqué au terme quadratique $\mathbf{F}_2^p(\mathcal{T})$ et au terme cubique $\mathbf{F}_3^p(\mathcal{T})$ de l'équation 3.7. Le premier se décompose en une force liée au déplacement du sommet concerné, une force liée aux déplacements des arêtes partant du sommet, et une force liée aux déplacements des triangles adjacents au sommet :

$$\mathbf{F}_2^p(\mathcal{T}) = \mathbf{F}^p_{\text{sommet p}}(\mathbf{U}_p) + \sum_{j\neq p} \mathbf{F}^p_{\text{aréte (p,j)}}(\mathbf{U}_p,\mathbf{U}_j) + \sum_{\substack{j,k\neq p \\ j<k}} \mathbf{F}^p_{\text{face (p,j,k)}}(\mathbf{U}_j,\mathbf{U}_k) \tag{3.9}$$

Il apporte donc des vecteurs de rigidité sur les sommets, les arêtes et les faces du maillage.
Le second se décompose en quatre parties, respectivement liées aux déplacements du sommet, des arêtes adjacentes au sommet, des faces partageant ce sommet, et enfin du tétraèdre lui-même :

$$\begin{aligned} \mathbf{F}_3^p(\mathcal{T}) = & \mathbf{F}^p_{\text{sommet p}}(\mathbf{U}_p) + \sum_{j\neq p} \mathbf{F}^p_{\text{arête (p,j)}}(\mathbf{U}_p,\mathbf{U}_j) + \sum_{\substack{j,k\neq p \\ j<k}} \mathbf{F}^p_{\text{face (p,j,k)}}(\mathbf{U}_p,\mathbf{U}_j,\mathbf{U}_k) \\ & + \mathbf{F}^p_{\text{tétraèdre (p,j,k,l)}}(\mathbf{U}_j,\mathbf{U}_k,\mathbf{U}_l) \end{aligned} \tag{3.10}$$

Ce dernier terme apporte donc ses contributions sous la forme de scalaires de rigidité sur les sommets, les arêtes, les faces et le tétraèdre. Une fois ces trois termes de forces additionnés et les contributions de chaque tétraèdre accumulées, on obtient l'expression de la force élastique exercée sur chaque sommet du maillage :

$$
\begin{aligned}
\mathbf{F}^p = \; & \boxed{\begin{array}{ll} \multicolumn{2}{c}{\text{Contribution du sommet}} \\ \hline & 2\left[\mathcal{B}^{pp}\right]\mathbf{U}_p \\ + & \left[2\left(\mathbf{U}_p \otimes \mathbf{U}_p\right) + \left(\mathbf{U}_p.\mathbf{U}_p\right) Id_3\right]\mathcal{C}^{ppp} \\ + & 4\mathcal{D}^{pppp}\,\mathbf{U}_p\,\mathbf{U}_p^t\,\mathbf{U}_p \end{array}} \\
+ & \sum_{\text{arêtes}(p,j)} \boxed{\begin{array}{ll} \multicolumn{2}{c}{\text{Contribution d'une arête}} \\ \hline & 2\left[\mathcal{B}^{pj}\right]\mathbf{U}_j \\ + & 2\left[\left(\mathbf{U}_j \otimes \mathbf{U}_p\right) + \left(\mathbf{U}_j.\mathbf{U}_p\right) Id_3\right]\mathcal{C}^{ppj} + 2\left(\mathbf{U}_p \otimes \mathbf{U}_j\right)\mathcal{C}^{jpp} \\ & +2\left(\mathbf{U}_j \otimes \mathbf{U}_j\right)\mathcal{C}^{jjp} + \left(\mathbf{U}_j.\mathbf{U}_j\right)\mathcal{C}^{pjj} \\ + & 4\Big[\mathcal{D}^{jppp}\,\left(2\mathbf{U}_p\,\mathbf{U}_p^t\,\mathbf{U}_j + \mathbf{U}_j\,\mathbf{U}_p^t\,\mathbf{U}_p\right) + \mathcal{D}^{jjpp}\,\mathbf{U}_p\,\mathbf{U}_j^t\,\mathbf{U}_j \\ & +\left(\mathcal{D}^{jpjp} + \mathcal{D}^{pjjp}\right)\mathbf{U}_j\,\mathbf{U}_j^t\,\mathbf{U}_p + \mathcal{D}^{jjjp}\,\mathbf{U}_j\,\mathbf{U}_j^t\,\mathbf{U}_j\Big] \end{array}} \\
+ & \sum_{\text{faces}(p,j,k)} \boxed{\begin{array}{ll} \multicolumn{2}{c}{\text{Contribution d'une face}} \\ \hline & 2\left[\left(\mathbf{U}_k \otimes \mathbf{U}_j\right)\mathcal{C}^{jkp} + \left(\mathbf{U}_j \otimes \mathbf{U}_k\right)\mathcal{C}^{kjp} + \left(\mathbf{U}_j.\mathbf{U}_k\right)\mathcal{C}^{pjk}\right] \\ + & 4\Big[\left(\mathcal{D}^{pjkp} + \mathcal{D}^{jpkp}\right)\left(\mathbf{U}_j\,\mathbf{U}_k^t\,\mathbf{U}_p + \mathbf{U}_k\,\mathbf{U}_j^t\,\mathbf{U}_p\right) + 2\mathcal{D}^{jkpp}\,\mathbf{U}_p\,\mathbf{U}_j^t\,\mathbf{U}_k \\ & +\left(\mathcal{D}^{kjjp} + \mathcal{D}^{jkjp}\right)\mathbf{U}_j\,\mathbf{U}_j^t\,\mathbf{U}_k + \mathcal{D}^{jjkp}\,\mathbf{U}_k\,\mathbf{U}_j^t\,\mathbf{U}_j \\ & +\left(\mathcal{D}^{jkkp} + \mathcal{D}^{kjkp}\right)\mathbf{U}_k\,\mathbf{U}_k^t\,\mathbf{U}_j + \mathcal{D}^{kkjp}\,\mathbf{U}_j\,\mathbf{U}_k^t\,\mathbf{U}_k\Big] \end{array}} \\
+ & \sum_{\text{tétraèdres}(p,j,k,l)} \boxed{\begin{array}{l} \multicolumn{1}{c}{\text{Contribution d'un tétraèdre}} \\ \hline 4\Big[\left(\mathcal{D}^{jklp} + \mathcal{D}^{kjlp}\right)\mathbf{U}_l\,\mathbf{U}_j^t\,\mathbf{U}_k + \left(\mathcal{D}^{jlkp} + \mathcal{D}^{ljkp}\right)\mathbf{U}_k\,\mathbf{U}_j^t\,\mathbf{U}_l \\ +\left(\mathcal{D}^{kljp} + \mathcal{D}^{lkjp}\right)\mathbf{U}_j\,\mathbf{U}_k^t\,\mathbf{U}_l\Big] \end{array}}
\end{aligned}
\tag{3.11}
$$

où les sommes se font sur les ensembles des arêtes, faces et tétraèdres adjacents au sommet concerné.

Une structure de données adéquate répartit les informations de rigidité sur les primitives géométriques du maillage, comme indiqué dans le tableau 3.1.

Répartition des informations de rigidité	Tenseurs	Vecteurs	Scalaires
Sommet $\mathbf{V}_p$	$\mathcal{B}^{pp}$	$\mathcal{C}^{ppp}$	$\mathcal{D}^{pppp}$
Arête $\mathbf{E}_{pj}$	$\mathcal{B}^{pj}$	$\mathcal{C}^{ppj}$ $\mathcal{C}^{jpp}$ $\mathcal{C}^{jjp}$ $\mathcal{C}^{pjj}$	$\mathcal{D}^{jppp}$ $\mathcal{D}^{jjjp}$ $\mathcal{D}^{jpjp}$ $\mathcal{D}^{pjjp}$ $\mathcal{D}^{jjpp}$
Face $\mathbf{F}_{pjk}$		$\mathcal{C}^{jkp}$ $\mathcal{C}^{kjp}$ $\mathcal{C}^{pjk}$	$\mathcal{D}^{jkpp}$ $\mathcal{D}^{jpkp}$ $\mathcal{D}^{pjkp}$ $\mathcal{D}^{jjkp}$ $\mathcal{D}^{jkjp}$ $\mathcal{D}^{kjjp}$ $\mathcal{D}^{kkjp}$ $\mathcal{D}^{kjkp}$ $\mathcal{D}^{jkkp}$
Tétraèdre $\mathbf{T}_{pjkl}$			$\mathcal{D}^{jklp}$ $\mathcal{D}^{jlkp}$ $\mathcal{D}^{kjlp}$ $\mathcal{D}^{kljp}$ $\mathcal{D}^{ljkp}$ $\mathcal{D}^{lkjp}$

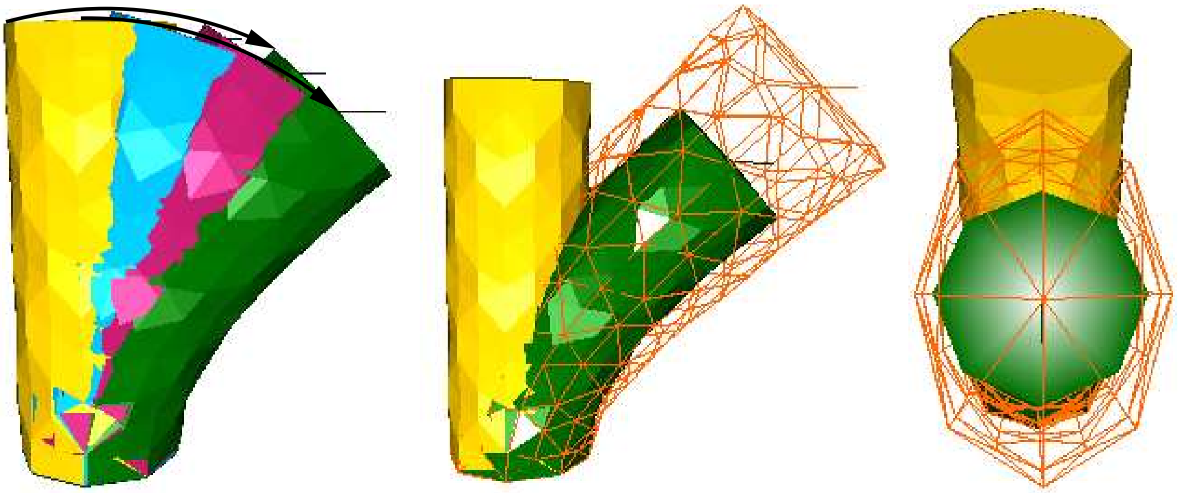
TAB. 3.1 – *Stockage des informations de rigidité sur les composantes géométriques du maillage.*

Le schéma de résolution de ce nouveau modèle déformable est presque identique à celui décrit dans le cas linéaire. Les seules différences sont dues à l'augmentation des informations de rigidité et à la complexité supérieure des équations permettant de calculer les forces :

- Tout d'abord le stockage des informations de rigidité nécessite l'ajout d'une structure de face (ou de triangle) à l'intérieur du maillage.
- Ensuite, le calcul des forces imposera de parcourir, en plus des sommets et des arêtes, les faces et les tétraèdres du maillage.
- Enfin, lors des éventuels changements de topologie du maillage, plus d'informations de rigidité devront être mises à jour. Par exemple si l'on retire un tétraèdre, il faudra modifier les données de 4 sommets, 6 arêtes et 4 faces, ce qui représente (cf. tableau 3.1) :

$$\begin{gathered} 4 * (1 \text{ tenseur} + 1 \text{ vecteur} + 1 \text{ scalaire}) \\ +6 * (1 \text{ tenseur} + 4 \text{ vecteurs} + 5 \text{ scalaires}) \\ +4 * (3 \text{ vecteurs} + 9 \text{ scalaires}) \\ = 280 \text{ réels} \end{gathered}$$

Afin de mettre en évidence l'apport du modèle masse-tenseur "grands déplacements", nous avons renouvelé l'expérience proposée au paragraphe 3.2 sur le cylindre (figure 3.2). On peut ainsi apprécier l'avantage du modèle non-linéaire, qui se déforme de manière beaucoup plus réaliste. En effet, on constate sur la figure 3.3(a) que les points peuvent maintenant suivre des trajectoires non rectilignes, permettant au cylindre de se courber naturellement. Si on compare ce résultat avec le modèle linéaire (figures 3.3(b) et 3.3(c)), on se rend compte de l'erreur commise par ce dernier dès que l'on sort de l'hypothèse des petits déplacements. Le modèle non-linéaire permet au cylindre de se plier sans que sa longueur n'augmente ou que sa section ne se déforme. On peut aussi apprécier l'avantage qu'il présente en terme de conservation du volume.



(a) Déformations successives du cylindre utilisant le modèle masse-tenseur "grands déplacements". Les flèches indiquent les trajectoires de certains points.

(b) Comparaison des déformations du modèle linéaire (rendu en fil de fer) et du modèle "grands déplacements" (en rendu plein).

(c) Même expérience que (b), mais vu de dessus.

FIG. 3.3 – *Application d'une force sur la face supérieure du cylindre (la face inférieure étant fixée). Comparaison entre le modèle linéaire (rendu en fil de fer) et le modèle "grands déplacements (en rendu plein).*

Par contre, ce modèle en grands déplacements présente un inconvénient. En plus d'être invariant par translation et par rotation, le tenseur de déformation de Green-St Venant E s'avère être invariant par symétrie.

En effet, considérons un plan $\mathcal{P}$ défini par un point $\mathbf{O}$ et un vecteur normal $\vec{\mathbf{N}}$. Alors la symétrie $\mathcal{S}_{\mathcal{P}}$ par rapport au plan $\mathcal{P}(\mathbf{O},\vec{\mathbf{N}})$ peut s'écrire :

$$\mathcal{S}_{\mathcal{P}}(\mathbf{X}) = \mathbf{X} - 2(\vec{\mathbf{N}} \otimes \vec{\mathbf{N}})(\mathbf{X} - \mathbf{O}) \tag{3.12}$$

On peut alors exprimer le gradient du champ de déplacements, et vérifier qu'il engendre un tenseur de déformation E nul :

$$
\begin{aligned}
\mathbf{U} &= \mathcal{S}_{\mathcal{P}}(\mathbf{X}) - \mathbf{X} = -2(\vec{\mathbf{N}} \otimes \vec{\mathbf{N}})(\mathbf{X} - \mathbf{O}) \\
\nabla \mathbf{U} &= -2(\vec{\mathbf{N}} \otimes \vec{\mathbf{N}}) \\
E &= \frac{1}{2}\big(-4(\vec{\mathbf{N}} \otimes \vec{\mathbf{N}}) + 4(\vec{\mathbf{N}} \otimes \vec{\mathbf{N}})(\vec{\mathbf{N}} \otimes \vec{\mathbf{N}})\big) \\
&= \frac{1}{2}\big(-4(\vec{\mathbf{N}} \otimes \vec{\mathbf{N}}) + 4\vec{\mathbf{N}}(\vec{\mathbf{N}}^t\vec{\mathbf{N}})\vec{\mathbf{N}}^t\big) \\
&= \frac{1}{2}\big(-4(\vec{\mathbf{N}} \otimes \vec{\mathbf{N}}) + 4(\vec{\mathbf{N}} \otimes \vec{\mathbf{N}})\big) = 0
\end{aligned}
$$

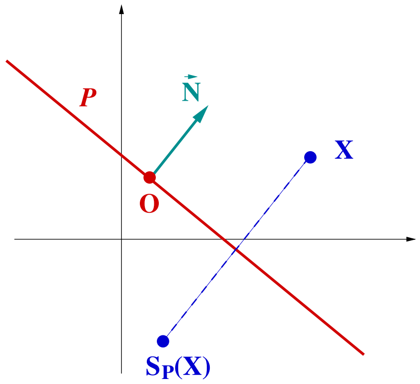


FIG. 3.4 – *Symétrie par rapport à un plan.*

Cela signifie qu'un tétraèdre pourrait se retourner et se retrouver dans une nouvelle position d'équilibre. En fait, il y a très peu de chance de rencontrer ce cas dans un maillage, car les tétraèdres adjacents s'empêchent mutuellement de se retourner. Mais nous avons observé ce phénomène sur la surface des maillages lorsque certains tétraèdres sont soumis à de fortes contraintes, par exemple lors d'un contact lorsque l'on impose des déplacements à certains triangles de la surface (la modélisation des contacts est traitée au chapitre suivant, paragraphe 4.2). Dans ces conditions extrêmes, on peut difficilement garantir que le tétraèdre ne va pas se retourner, mais on peut faire en sorte qu'il n'y trouve pas une position d'équilibre stable en pénalisant sa variation de volume.

Nous avons vu dans ce paragraphe que l'utilisation de l'élasticité de St Venant-Kirchhoff améliore énormément la conservation du volume de l'objet simulé, en évitant les dilatation irréalistes entraînées par les rotations locales. Motivé par le problème du retournement des tétraèdres que nous venons d'évoquer, nous avons voulu aller plus loin en intégrant à notre modèle des contraintes d'incompressibilité.

3.3.2 Contraintes d'incompressibilité

Les tissus vivants, qui sont composés essentiellement d'eau, peuvent être considérés comme quasiment incompressibles. Cette propriété est particulièrement difficile à modéliser et conduit la plupart du temps à des problèmes d'instabilité. C'est le cas avec le modèle de St Venant-Kirchhoff, car le matériau simulé atteint un comportement incompressible lorsque la constante de Lamé λ tend vers l'infini. Mais prendre une valeur trop grande pour λ nous forcerait à diminuer le pas de temps et donc augmenter le temps de convergence. Nous avons donc décidé d'ajouter une contrainte d'incompressibilité externe au modèle d'élasticité. Comme nous sommes obligés, lors du calcul des forces élastiques, de parcourir tous les tétraèdres du maillage, il nous a semblé que le plus simple, mais aussi le plus efficace, serait d'appliquer une contrainte géométrique sur chaque tétraèdre.

Nous avons donc choisi de pénaliser la variation de volume de chaque tétraèdre en appliquant sur chacun de ses sommets, une force dirigée selon la normale à la face opposée. La norme de cette force étant proportionnelle au carré de la variation relative de volume et le sens étant donné par le signe de cette variation :

$$\mathbf{F}^p_{\text{incomp}} = \gamma \; sign(V - V_0) \left(\frac{V - V_0}{V_0}\right)^2 \vec{\mathbf{N}}_p$$

Ces forces agissent comme une augmentation de la pression à l'intérieur du tétraèdre. Cette méthode est relativement proche des multiplicateurs de Lagrange, qui sont souvent utilisés pour résoudre les problèmes de minimisation d'énergie sous contraintes.

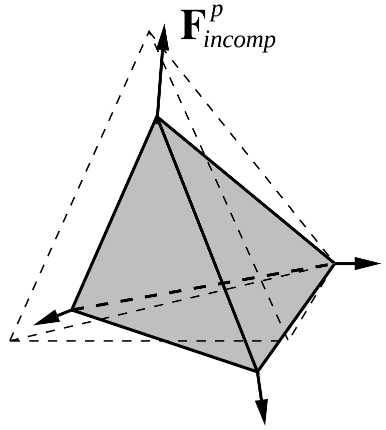


FIG. 3.5 – *Pénalisation de la variation de volume. La position au repos est représentée en pointillés.*

En effet, pour formuler le problème de minimisation de l'énergie élastique en incluant la contrainte d'incompressibilité, nous devons ajouter un terme de la forme $\boldsymbol{\lambda} \; f(V - V_0)$, pénalisant la variation de volume de chaque tétraèdre. Lors de la minimisation de cette énergie, nous sommes donc amenés à dériver le volume du tétraèdre par rapport à la position de ses sommets. Ce volume s'exprime, en utilisant le produit mixte, sous la forme :

$$\begin{aligned} 6\,V &= (\mathbf{P}_1 - \mathbf{P}_0) \,.\, (\mathbf{P}_2 - \mathbf{P}_0) \wedge (\mathbf{P}_3 - \mathbf{P}_0) \\ &= \mathbf{P}_1 \,.\, \mathbf{P}_2 \wedge \mathbf{P}_3 - \mathbf{P}_1 \,.\, \mathbf{P}_2 \wedge \mathbf{P}_0 - \mathbf{P}_1 \,.\, \mathbf{P}_0 \wedge \mathbf{P}_3 - \mathbf{P}_0 \,.\, \mathbf{P}_2 \wedge \mathbf{P}_3 \end{aligned} \tag{3.14}$$

Si on dérive ce volume par rapport à la position du sommet $\mathbf{P}_1$, on obtient :

$$\begin{aligned} \frac{\partial V}{\partial \mathbf{P}_1} &= \frac{1}{6} (\mathbf{P}_2 \wedge \mathbf{P}_3 + \mathbf{P}_3 \wedge \mathbf{P}_0 + \mathbf{P}_0 \wedge \mathbf{P}_3) \\ &= \frac{\mathcal{A}_1}{3} \, \vec{\mathbf{N}}_1 \end{aligned} \tag{3.15}$$

où $\mathcal{A}_1$ est l'aire de la face $(\mathbf{P}_0,\mathbf{P}_2,\mathbf{P}_3)$. On retrouve bien que la pénalisation de la variation de volume entraîne une force en chaque sommet du tétraèdre, dont la direction est perpendiculaire à la face opposée.

3.3.3 Résultats

La figure 3.6 présente la comparaison du modèle masse-tenseur non-linéaire avec son homologue linéaire. Les modèles déformables ont été fixés au centre de leur face antérieure : cette zone correspond au *hile* hépatique, c'est-à-dire l'endroit ou les différents arbres vasculaires (artériel, veineux et biliaire) pénètrent dans le parenchyme hépatique. Nous utilisons un coefficient μ de l'ordre de $10^4 - 10^5 \; kg/cm^2$ (voir paragraphe 5.3.1 sur les propriétés biomécaniques du foie), alors que les valeurs de λ sont issues du compromis entre un comportement le plus incompressible possible (λ très grand) et un temps de calcul correct. En général, cela nous mène à des valeurs de λ comprises entre 2μ et 10μ. Nous avons alors appliqué des forces sur le lobe droit du foie. Si on utilise le modèle linéaire, la partie droite du foie subit une importante (et irréaliste) augmentation de volume, alors qu'avec le modèle non-linéaire, le lobe droit est capable de subir une rotation partielle, ce qui donne une impression de déformation beaucoup plus proche de la réalité.

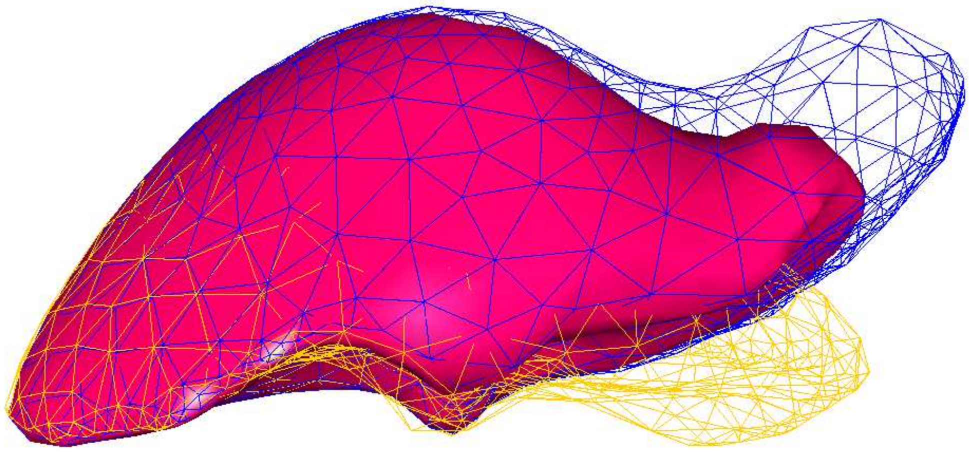
Lorsque l'on ajoute la contrainte d'incompressibilité décrite au paragraphe précédent, on obtient des variations de volume encore plus faibles (voir table 3.2). L'ajout de cette contrainte a aussi pour effet de stabiliser le comportement dans les zones de plus forte déformation.

Variations de volume (%)	Linéaire	Non-linéaire	Non-linéaire incomp.
Cylindre gauche | milieu | droit	7 | 28 | 63	0.3 | 1 | 2	0.2 | 0.5 | 1
Foie	9	1.5	0.7

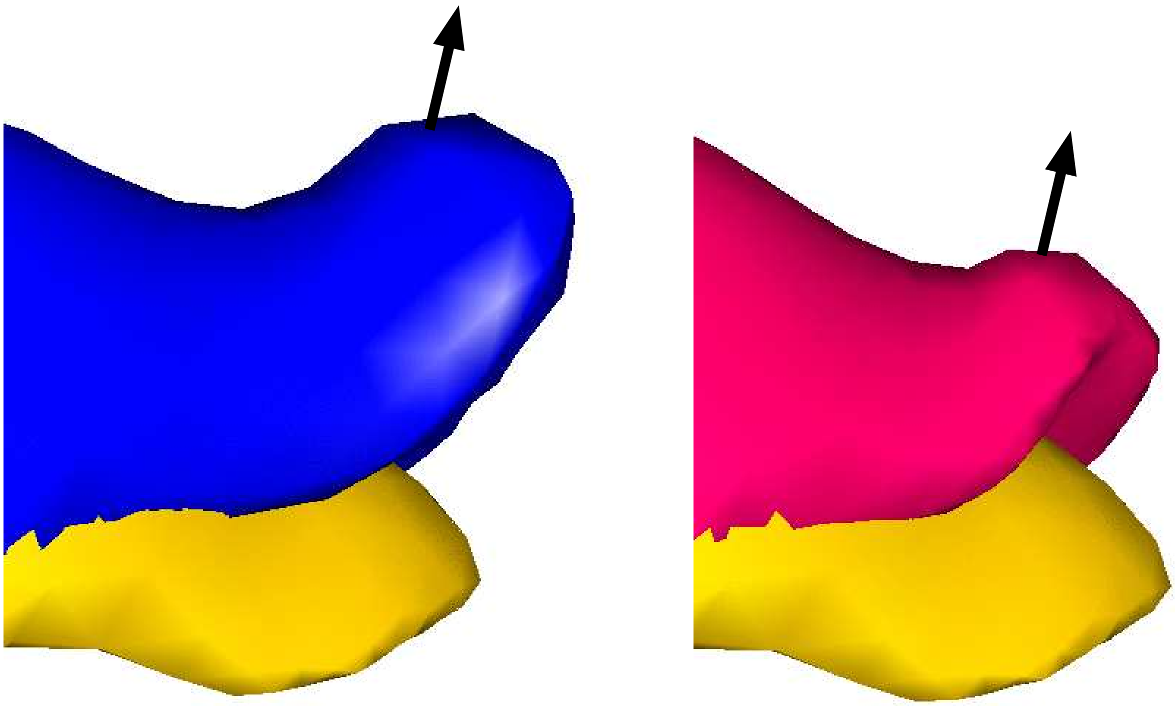
TAB. 3.2 – *Résultats de variations de volume. Pour le cylindre, les appellations gauche, milieu et droit font références aux différents stades de déformation des figures 3.2 et 3.3(a).*

Enfin, nous présentons un exemple de simulation d'un geste classique de chirurgie laparoscopique sur le foie. Le chirurgien opère à l'aide de deux instruments. Celui de droite est une pince avec laquelle il va saisir une zone du lobe droit du foie afin de le mettre sous tension, pendant qu'il va découper le parenchyme hépatique avec l'instrument de gauche. Au fur et à mesure de la découpe, le chirurgien écarte la partie du foie qu'il veut supprimer afin de faciliter la suite de la découpe. Cette partie peut alors subir un important mouvement de rotation qui sera fidèlement simulé par notre modèle non-linéaire (figure 3.7).

Évidemment, le temps de calcul des forces pour ce modèle non-linéaire est nettement plus long que dans le cas linéaire. D'abord parce qu'il nous faut parcourir, en plus des sommets et des arêtes, tous les triangles et tous les tétraèdres du maillage. Ensuite parce que les expressions des forces sont nettement plus compliquées (voir équation 3.11). Après avoir factorisé au maximum les calculs, nous avons obtenu une fréquence de simulation 5 fois inférieure à celle du modèle linéaire, soit environ 9 Hz sur un pentium PIII 600 MHz pour un modèle de foie comportant 6342 tétraèdres (ce qui correspond à 1394 sommets), alors qu'avec le modèle linéaire nous atteignons la fréquence de 45 Hz. Ce temps de calcul permet néanmoins d'obtenir une fréquence de 30 Hz avec un maillage comportant environ 2000 tétraèdres. C'est le cas en particulier pour l'exemple de la figure 3.7, où seulement le tiers droit du foie est déformable. Nous avons essayé d'améliorer le temps de calcul par



(a) Comparaison des déformations du modèle linéaire (en rendu fil de fer) et du modèle non-linéaire (en rendu de surface). La position au repos est représentée par maillage fil de fer du bas (jaune).



(b) Gros plan sur la zone de plus forte déformation. La forme au repos est représentée par le modèle du bas (jaune). On constate une forte différence vis à vis de la conservation du volume entre le modèle linéaire (à gauche) et le modèle grands déplacements (à droite).

FIG. 3.6 – *Comparaison entre le modèle linéaire et le modèle non-linéaire pour simuler les déformation d'un foie.*

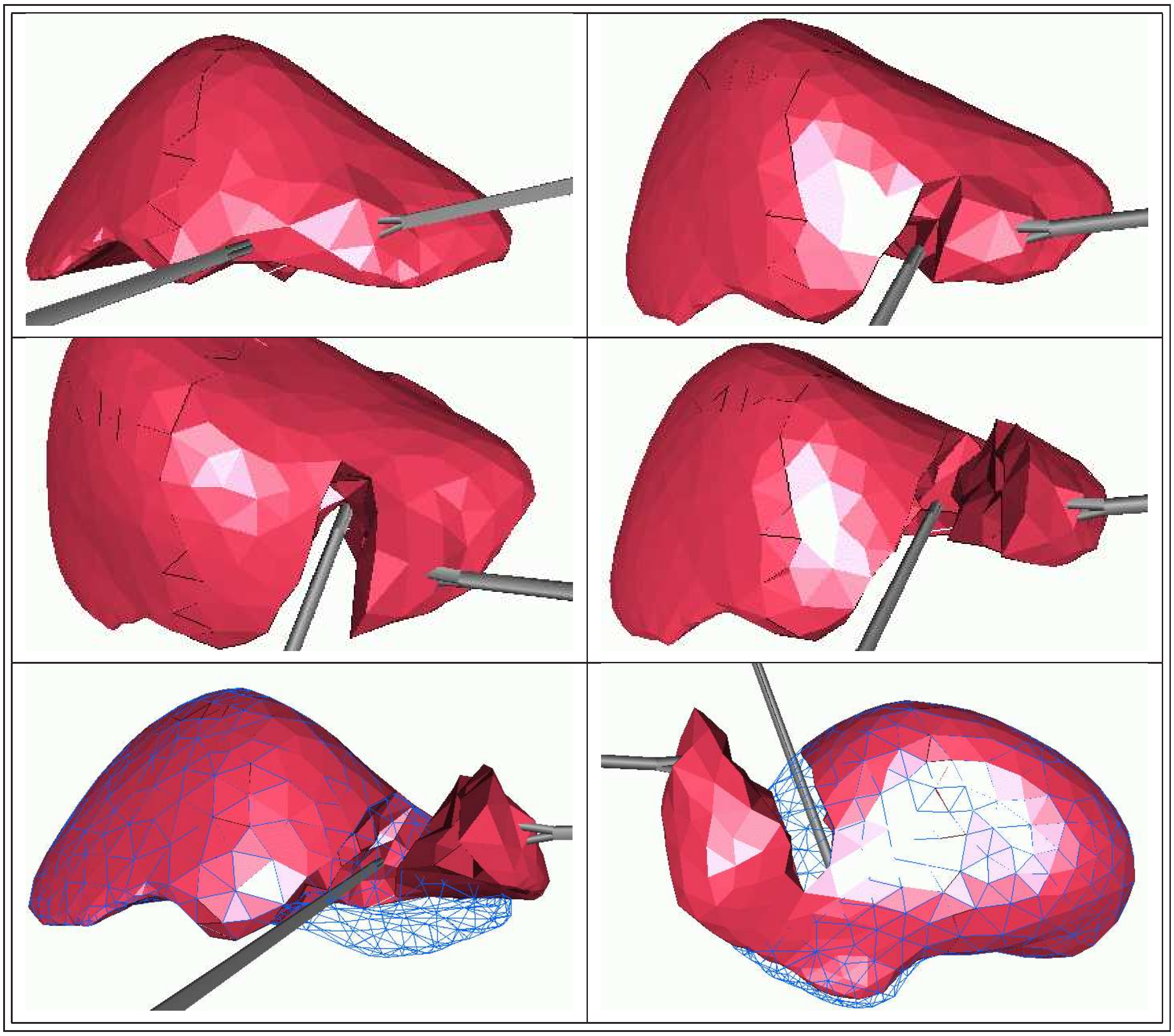


Fig. 3.7 – *Simulation de chirurgie laparoscopique du foie en utilisant le modèle masse-tenseur "grands déplacements". Sur les deux dernières images, la forme au repos est représentée en rendu fil de fer.*

quelques "bricolages" plus ou moins justifiables. Ainsi, nous avons augmenté la fréquence jusqu'à 14 Hz en ne recalculant les termes les plus coûteux (notamment les contributions des tétraèdres, cf. équation 3.11) que si les positions ont suffisamment changés d'un pas de temps à l'autre. Mais ce genre de méthode d'optimisation n'étant pas très satisfaisante sur le plan théorique, nous avons mis en œuvre une autre solution, que nous allons exposer maintenant.

3.3.4 Optimisation : non-linéaire adaptatif

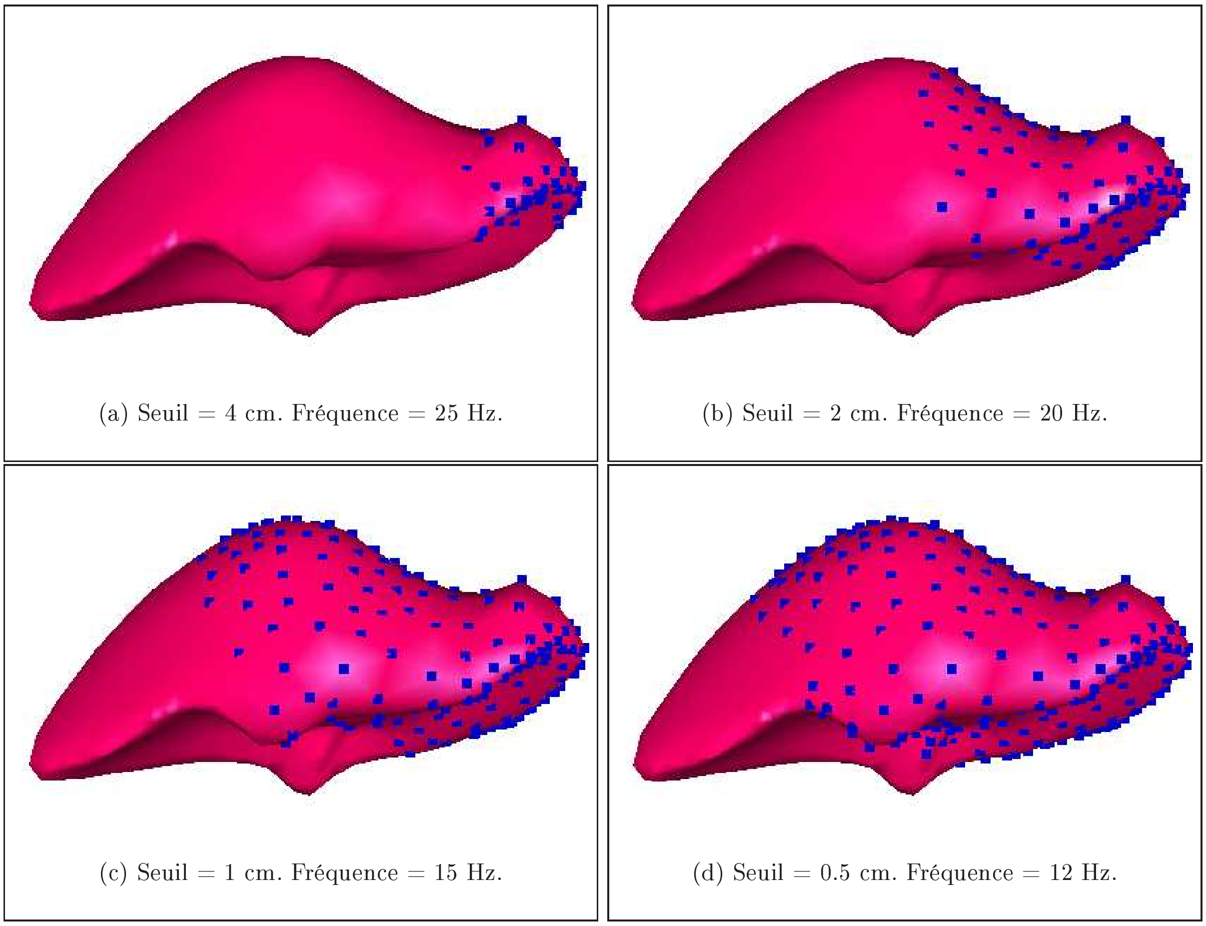
Le premier terme de la force élastique exercée sur un sommet du maillage correspond exactement avec celle du modèle linéaire. On peut donc considérer que le modèle de St Venant-Kirchhoff est un modèle linéaire auquel on a ajouté des termes correctifs de degrés deux et trois lui permettant de garantir une invariance par rotation. Lorsque le déplacement du sommet est "petit", ces termes d'ordres élevés deviennent négligeables, et comme ils sont coûteux à calculer, il semble alors évident que nous aurions tout intérêt à ne pas en tenir compte.

Nous introduisons donc ici ce que nous appelons un modèle non-linéaire adaptatif. L'idée de base est de se contenter d'utiliser de l'élasticité linéaire pour tous les nœuds dont les déplacements sont inférieurs à une valeur seuil, et de ne calculer les termes non-linéaires que pour les sommets soumis à de grands déplacements. Le modèle devient donc adaptatif dans la mesure où son comportement biomécanique s'adapte à la déformation locale.

Nous avons donc modifié le fonctionnement de notre boucle de simulation pour rendre le modèle adaptatif. Pour calculer la force élastique exercée sur chaque sommet, nous parcourons successivement les sommets, les arêtes, les triangles et les tétraèdres du maillage, et chaque primitive géométrique apporte sa contribution à la force. Pour le modèle non-linéaire adaptatif, nous commençons par parcourir les sommets du maillage. Pour chaque sommet, nous calculons la partie linéaire de la force. Ensuite, nous testons si le déplacement du sommet est supérieur au seuil. Si il l'est, on calcule la partie non-linéaire. Par contre si le déplacement est inférieur au seuil, on ne calcule pas la partie non-linéaire et on place sur le sommet un drapeau indiquant qu'il doit être traité comme un "sommet linéaire". On parcourt ensuite chaque arête, qui commence par apporter sa contribution linéaire. Elle vérifie ensuite que ses sommets ne portent pas de drapeau avant de calculer et de leur ajouter la partie non-linéaire de la force. Les triangles et les tétraèdres ne contribuant que pour la partie non-linéaire de la force, ils ne la calculeront que pour les sommets ne portant pas de drapeau. Enfin, tous les drapeaux sont retirés après le calcul des forces.

Lorsqu'il n'est soumis à aucune déformation, le modèle adaptatif se comporte comme le modèle linéaire. Mais en raison de sa structure plus complexe, du parcours d'un plus grand nombre de primitives géométriques et de la présence de nombreux tests (pour voir si les sommets portent un drapeau), il est moins performant : on obtient une fréquence de 30 Hz, alors que le modèle linéaire atteint 45 Hz et que le non-linéaire est limité à 9 Hz. Cette fréquence de 30 Hz va bien entendu diminuer au fur et à mesure que le nombre de sommets non-linéaires va augmenter.

La figure 3.8 présente un modèle de foie non-linéaire adaptatif. La même déformation (déplacement de trois sommets d'environ 5 cm) a été appliquée dans les quatre cas, mais avec des valeurs différentes du seuil. On observe des variations de forme entre les différents cas, et si le seuil de 4 cm de la figure 3.8(a) s'avère un peu trop grand (on observe une dilatation du modèle à la frontière des points bleus), il semble que les valeurs comprises entre 1 et 2 centimètres donnent une bonne approximation de la déformation. Par contre, il apparaît, dans ce cas, inutile de prendre un seuil aussi bas que 0.5 centimètres, le gain visuel en précision étant infime. On préférera gagner en fréquence de simulation.



(a) Seuil = 4 cm. Fréquence = 25 Hz.

(b) Seuil = 2 cm. Fréquence = 20 Hz.

(c) Seuil = 1 cm. Fréquence = 15 Hz.

(d) Seuil = 0.5 cm. Fréquence = 12 Hz.

FIG. 3.8 – *Déformation du modèle non-linéaire adaptatif pour différentes valeurs du seuil. Les sommets portant un carré bleu sont non-linéaires, les autres sont linéaires.*

Il est évident que pour une valeur donnée du seuil, le nombre de sommets non-linéaires va augmenter avec l'intensité de la déformation. Donc, plus la déformation sera grande, plus la fréquence de simulation diminuera. Mais un grand nombre de gestes chirurgicaux, surtout effectués en laparoscopie, ne provoquent que de petits déplacements des organes, ou bien des déplacements importants sur une partie limitée d'un organe. C'est le cas par exemple lorsque le chirurgien découpe un morceau du foie (voir figure 3.7). La figure 3.9 montre la même séquence de simulation mais en utilisant cette fois le modèle adaptatif. On s'aperçoit que les sommets non-linéaires sont concentrés sur la partie du foie que l'on est en train de découper, ce qui lui permet de suivre, sans se dilater, le mouvement de rotation que lui impose l'instrument de droite. Par contre, tout le reste du foie, qui est très peu déformé, reste linéaire. Par rapport au modèle entièrement non-linéaire, la fréquence de simulation est passée de 20 Hz à une valeur comprise entre 50 et 80 Hz suivant le nombre de sommets considérés comme non-linéaires (la fréquence minimale de 50 Hz correspond à la fin de l'opération, lorsque tous les sommets appartenant au morceau découpé sont non-linéaires).

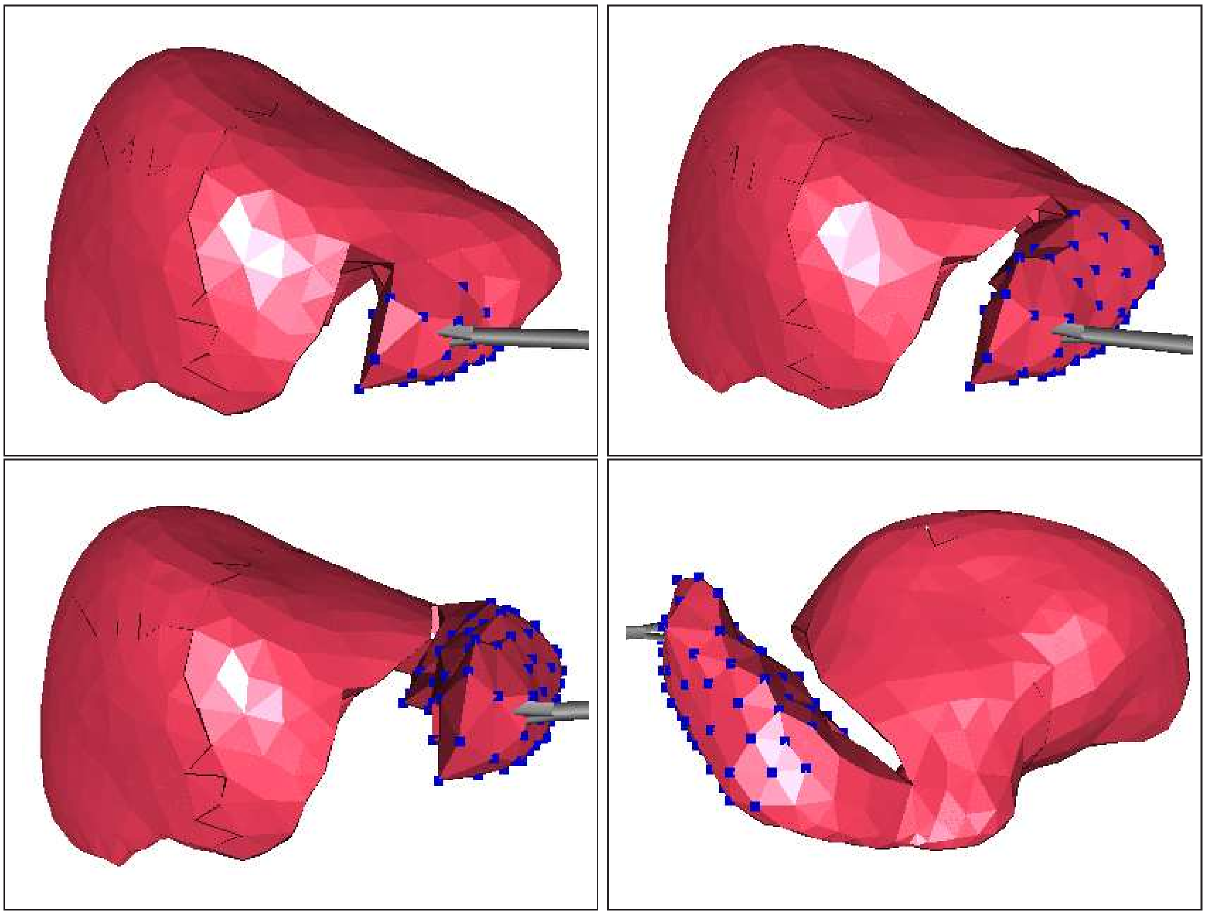


Fig. 3.9 – *Simulation de découpe du foie avec le modèle adaptatif. Le modèle déformable comporte environ 2500 tétraèdres. Pour un seuil de 2 cm, la fréquence de simulation varie entre 50 et 80 Hz.*

Ce modèle adaptatif permet donc de profiter des avantages de l'élasticité en grands déplacements avec une augmentation du temps de calcul qui reste raisonnable dans la plupart des applications. On peut régler la valeur du seuil entre petits et grands déplacements en fonction du réalisme et de la fréquence de simulation que l'on désire atteindre. On pourrait même aller encore plus loin en rendant cette adaptation automatique. En effet, il serait relativement simple de mettre en place un processus de correction du seuil qui en adapterait la valeur afin de maintenir une fréquence de simulation donnée.

3.3.5 L'anisotropie en grands déplacements

Le modèle déformable masse-tenseur non-linéaire que nous avons développé sur la base de l'élasticité en grands déplacements de St Venant-Kirchhoff apporte une amélioration en réalisme indéniable par rapport au modèle linéaire. De plus, cette amélioration se fait au prix d'un surcroît de calcul raisonnable grâce à l'utilisation du modèle adaptatif que nous venons de présenter. Nous avons donc voulu transmettre à ce nouveau modèle toutes les améliorations que nous avions développées dans le cas linéaire, en particulier la possibilité de modéliser des matériaux anisotropes. En ce qui concerne l'anisotropie de surface, le modèle de membrane masse-ressort s'applique ici sans aucune modification. Par contre, l'isotropie transverse demande un peu plus de travail.

Lors de la présentation de l'élasticité transversalement isotrope au paragraphe 2.5.1, nous avions distingué des termes d'énergie correspondant au cisaillement et à l'étirement anisotrope. Ce dernier étant à nos yeux le plus important, nous allons maintenant généraliser la notion d'étirement anisotrope au modèle d'élasticité de St Venant-Kirchhoff. Nous rappelons que la contribution de l'étirement anisotrope à l'énergie élastique s'exprime en fonction des coefficients de Lamé et de la direction d'anisotropie $\mathbf{a}_0$ par l'équation :

$$\Delta W_{\text{Étir\_aniso1}} = \left(\frac{\Delta\lambda}{2} + \Delta\mu\right)\left(\mathbf{a}_0^t \; E \; \mathbf{a}_0\right)^2 \tag{3.16}$$

Lorsque l'on traduit cette énergie pour chaque tétraèdre du maillage, on obtient une équation similaire à celle de l'énergie isotrope (voir annexe B.3 pour plus de détails) :

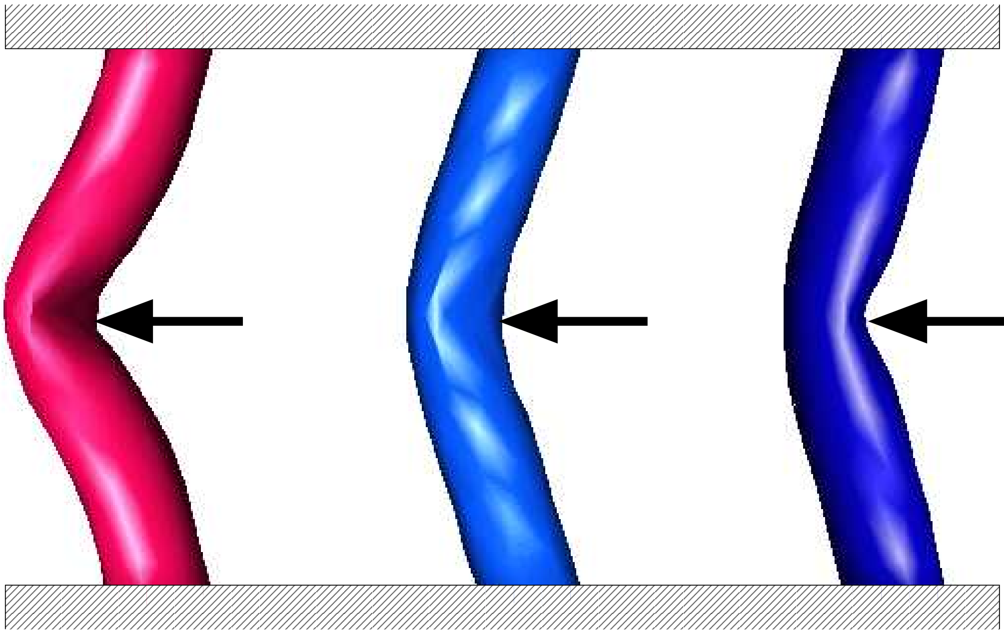
$$\begin{aligned}\Delta W_{\text{Étir\_aniso1}} = & \sum_{j,k} \mathbf{U}_j^t \left[\mathcal{B}_{\mathbf{a}_0}^{jk}\right] \mathbf{U}_k + \sum_{j,k,l} \left(\mathbf{U}_j \,.\, \mathcal{C}_{\mathbf{a}_0}^{jkl}\right) (\mathbf{U}_k \,.\, \mathbf{U}_l) \\ & + \sum_{j,k,l,m} \mathcal{D}_{\mathbf{a}_0}^{jklm} (\mathbf{U}_j \,.\, \mathbf{U}_k)(\mathbf{U}_l \,.\, \mathbf{U}_m)\,.\end{aligned} \tag{3.17}$$

$$\text{où} \begin{cases} \mathcal{B}_{\mathbf{a}_0}^{jk} = \left(\frac{\Delta\lambda}{2} + \Delta\mu\right)(\mathbf{a}_0 \otimes \mathbf{a}_0)\,(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k)\,(\mathbf{a}_0 \otimes \mathbf{a}_0) \\ \mathcal{C}_{\mathbf{a}_0}^{jkl} = \left(\frac{\Delta\lambda}{2} + \Delta\mu\right)(\mathbf{a}_0 \otimes \mathbf{a}_0)\,(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k)\,(\mathbf{a}_0 \otimes \mathbf{a}_0)\,\boldsymbol{\alpha}_l \\ \mathcal{D}_{\mathbf{a}_0}^{jklm} = \left(\frac{\Delta\lambda}{2} + \Delta\mu\right)(\mathbf{a}_0 \,.\, \boldsymbol{\alpha}_j)\,(\mathbf{a}_0 \,.\, \boldsymbol{\alpha}_k)\,(\mathbf{a}_0 \,.\, \boldsymbol{\alpha}_l)\,(\mathbf{a}_0 \,.\, \boldsymbol{\alpha}_m) \end{cases} \tag{3.18}$$

Il faut ajouter cette contribution anisotrope à l'énergie isotrope pour obtenir l'énergie de déformation d'un matériau non-linéaire transversalement isotrope. Pour cela il suffit donc d'additionner les tenseurs, vecteurs et scalaires de rigidité isotropes et anisotropes. La dérivation des forces, l'accumulation des informations de rigidité sur le maillage, ainsi que le calcul des forces élastiques exercées sur chaque sommet sont parfaitement identiques au cas isotrope.

Toutes les propriétés mises en évidence lorsque nous avons étudié l'anisotropie sur le modèle d'élasticité linéaire restent les mêmes. Ainsi, comme nous ne prenons ici en compte que le terme d'étirement anisotrope, le matériau simulé se comporte exactement comme s'il était isotrope lorsque les déformations se produisent dans le plan perpendiculaire à la direction d'anisotropie. Par contre le long de cette direction, le matériau est beaucoup plus raide.

Nous présentons néanmoins un nouvel exemple sur la figure 3.10, car cette expérience ne serait pas concluante en utilisant de l'élasticité linéaire. En effet, les rotations importantes subies par les modèles entraîneraient des dilatations qui viendraient masquer le résultat. Les trois cylindres sont fixés à leurs deux extrémités, au repos. On leur applique le même ensemble de forces. Le modèle de gauche est isotrope, il se déforme de la même manière dans toutes les directions. Ses conditions d'attachement aux extrémités lui donnent cette aspect courbe. Le modèle du milieu est anisotrope dans la direction de la longueur du cylindre. Son étirement étant beaucoup plus contraint, le matériau a tendance à se tendre afin de minimiser sa longueur le long de la direction d'anisotropie. C'est ce qui se produit lorsque l'on appuie sur une corde un peu élastique. Le modèle de droite est encore plus raide dans la direction privilégiée. On constate deux phénomènes intéressants. Tout d'abord, comme on s'y attendait, l'étirement du matériau est moins important. Mais on constate aussi que le cylindre a tendance à s'écraser à l'endroit où les forces ont été appliquées. En effet, sa grande anisotropie l'empêche de s'allonger, mais par contre sa rigidité relativement faible dans le plan orthogonal à la fibre provoque son écrasement. Ce comportement est assez intéressant car il correspond à ce que l'on obtiendrait avec un vaisseau sanguin : ses parois fibreuses lui apportent une bonne raideur longitudinale, mais sa structure creuse remplie de sang le prédispose à s'aplatir sous l'effet d'un pincement ou d'un contact.



$\lambda = 2.10^5\ kg/cm^2$	$\lambda = 2.10^5\ kg/cm^2$	$\lambda = 2.10^5\ kg/cm^2$
$\mu = 10^5\ kg/cm^2$	$\mu = 10^5\ kg/cm^2$	$\mu = 10^5\ kg/cm^2$
-	$\lambda^L = 10^6\ kg/cm^2$	$\lambda^L = 2.10^6\ kg/cm^2$
-	$\mu^L = 5.10^5\ kg/cm^2$	$\mu^L = 10^6\ kg/cm^2$

FIG. 3.10 – *Modélisation d'une structure tubulaire à l'aide de l'élasticité non-linéaire transversalement isotrope. Le modèle de gauche est isotrope, et les deux autres sont anisotropes, celui de droite étant deux fois plus raide dans la direction d'anisotropie (c'est-à-dire le long du cylindre).*

3.4 Les simplexes hyperélastiques

Dans le paragraphe précédant, nous avons développé un nouveau modèle déformable basé sur l'élasticité en grands déplacements de St Venant-Kirchhoff et la méthode des éléments finis. Nous l'avons présenté sous la forme d'une généralisation du modèle masse-tenseur défini au chapitre 2, dont la linéarité limitait la validité aux petits déplacements. Le principal apport du modèle en grands déplacements est sa propriété d'invariance par rotation qui permet d'éviter l'apparition de distorsions irréalistes lorsqu'une partie du modèle subit une rotation trop importante. Ce modèle est donc invariant par transformation rigide, c'est-à-dire translation et rotation. Cela signifie qu'il peut subir n'importe quel mouvement global de translation et de rotation tout en conservant une énergie et des contraintes internes nulles. Or nous avons suivi, jusqu'à présent, un raisonnement en déplacements, ce qui signifie que les forces internes sont calculées en fonction des déplacements des sommets du maillage. Dans le cas d'un mouvement rigide, les déplacements des sommets sont non nuls, et pourtant la force calculée en chaque sommet vaut bien zéro, ce qui n'est pas évident si on observe l'équation 3.11. Pourtant cette équation cache bien des termes qui vont s'annuler pour la composante rigide de la déformation. En conséquence, on ne peut s'empêcher de penser que ces termes sont calculés pour rien.

Cela donne donc envie d'intégrer implicitement la notion d'invariance par transformation rigide dans l'équation de l'énergie élastique, c'est-à-dire trouver une formulation de l'énergie qui ne dépende que du changement de forme du modèle. Or nous discrétisons notre problème par la méthode des éléments finis, ce qui veut dire que nous approximons l'énergie élastique sur un ensemble de sous-domaines définis par des formes géométriques simples. Nous sommes donc ramenés à chercher, sur chaque sous-domaine, une formulation de l'énergie qui ne dépende que du changement de forme de ce domaine. Considérons des sous-domaines de forme triangulaire (dans le cas d'un problème en dimension deux) et tétraédrique (en dimension trois). Nous savons que les formes de ces deux primitives géométriques sont parfaitement définies par les longueurs de leurs arêtes. Il semble alors possible d'exprimer l'énergie élastique sur chaque triangle (ou tétraèdre) uniquement en fonction des variations de longueur de ses arêtes.

Cette possibilité présente plusieurs avantages. Tout d'abord on ne raisonne plus par rapport à une position de repos (raisonnement en déplacements), mais par rapport à une forme de repos, ce qui semble plus naturel, et qui évitera les problèmes de précision qui apparaissent lorsque le modèle s'éloigne de sa position de repos. De plus les calculs seront simplifiés du fait qu'il n'est plus nécessaire de compenser les transformations rigides, l'invariance par transformation rigide étant intrinsèque à la formulation du modèle déformable.

Nous allons, dans ce paragraphe, appliquer ce raisonnement aux cas de l'élasticité en deux dimensions, puis en trois dimensions.

3.4.1 Élasticité 2D : triangle hyperélastique

Nous commençons par l'étude de l'élasticité en deux dimensions, et ceci pour plusieurs raisons. D'une part parce que les calculs sont plus simples dans ce cas et qu'il sera donc plus facile de présenter le raisonnement complet. De plus, ces calculs se généralisent presque directement au cas de l'élasticité en trois dimensions. D'autre part, utiliser l'élasticité 2D ne veut pas dire se restreindre aux problèmes 2D. On peut très bien modéliser une surface dans un espace à trois dimensions sous la forme d'une juxtaposition d'éléments plans (Zienkiewicz, 1977). On obtient ainsi un modèle de membrane qui peut s'avérer nécessaire pour la modélisation de certains organes du corps humain.

Notre but est donc d'exprimer l'énergie élastique de St Venant-Kirchhoff sur un élément fini linéaire triangulaire, uniquement en fonction des variations de longueur de ses arêtes.

3.4.1.1 Le tenseur de déformation

On considère un triangle $T^0 = (\mathbf{P}_0, \mathbf{P}_1, \mathbf{P}_2)$ du plan $\mathbb{R}^2$, caractérisé par les longueurs de ses arêtes (l_0, l_1, l_2) et ses angles aux sommets $(\theta_0, \theta_1, \theta_2)$. Ce triangle subit une déformation $\mathbf{\Phi}$ pour donner le triangle $T = (\mathbf{Q}_0, \mathbf{Q}_1, \mathbf{Q}_2)$ (voir figure 3.11).
Contrairement aux chapitres précédents, nous allons exprimer l'énergie potentielle élastique de cette

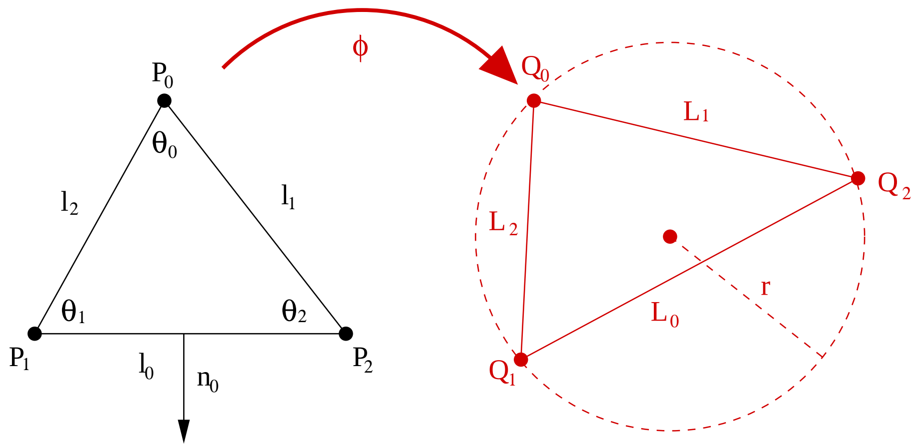


FIG. 3.11 – *Notations dans le triangle.*

déformation non plus à partir du tenseur de déformation E, mais à partir du tenseur de déformation de Cauchy-Green C :

$$C = \nabla\mathbf{\Phi}^t\nabla\mathbf{\Phi}. \tag{3.19}$$

De manière similaire à ce qui a été fait dans le chapitre 1.3 avec des tétraèdres, on considère l'élément fini obtenu en associant à ce triangle un ensemble de fonctions d'interpolation linéaires. On va ainsi pouvoir exprimer la déformation $\mathbf{\Phi}(\mathbf{X})$, puis son gradient, en tout point $\mathbf{X}$ du triangle par :

$$\mathbf{\Phi}(\mathbf{X}) = \sum_{j=0}^{2} \Lambda_j(\mathbf{X})\mathbf{Q}_j \implies \nabla\mathbf{\Phi} = \sum_{j=0}^{2} \boldsymbol{\alpha}_j \otimes \mathbf{Q}_j \quad \text{où} \quad \boldsymbol{\alpha}_j = \frac{l_j}{2\mathcal{A}}\mathbf{n}_j. \tag{3.20}$$

$\mathcal{A}$ est l'aire du triangle T_0, $\mathbf{n}_j$ est le vecteur unitaire perpendiculaire au segment $(\mathbf{P}_{j+1},\mathbf{P}_{j+2})$ [1], et les $\boldsymbol{\alpha}_j$ sont les vecteurs de forme du triangle.
Le tenseur de déformation C devient alors :

$$C = \sum_{j,k} (\mathbf{Q}_j \,.\, \mathbf{Q}_k)\,(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k)\,. \tag{3.21}$$

À ce stade, on peut remarquer que ce tenseur de déformation est invariant par translation et donc qu'il ne dépend pas du choix de l'origine du repère. Pour s'en convaincre, il suffit d'effectuer un changement d'origine en posant $\mathbf{Q}_j = \mathbf{Q}^*_j - \mathbf{T}$ et de se souvenir que $\sum_i \boldsymbol{\alpha}_i = \vec{0}$:

$$\begin{aligned} C \;=\; & \sum_{j,k} \left(\mathbf{Q}^*_j \,.\, \mathbf{Q}^*_k\right)(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k) \\ - \; & \underbrace{\Big(\sum_j \boldsymbol{\alpha}_j\Big)}_{=\vec{0}} \otimes \Big(\sum_k \boldsymbol{\alpha}_k(\mathbf{T} \,.\, \mathbf{Q}^*_k)\Big) - \Big(\sum_j (\mathbf{T} \,.\, \mathbf{Q}^*_k)\boldsymbol{\alpha}_k\Big) \otimes \underbrace{\Big(\sum_k \boldsymbol{\alpha}_k\Big)}_{=\vec{0}} \\ + \; & \|\mathbf{T}\|^2 \underbrace{\Big(\sum_j \boldsymbol{\alpha}_j\Big)}_{=\vec{0}} \otimes \underbrace{\Big(\sum_k \boldsymbol{\alpha}_k\Big)}_{=\vec{0}} \end{aligned}$$

On peut donc placer l'origine du repère au centre du cercle circonscrit au triangle T, dont le rayon vaut r (voir figure 3.11), de manière à exprimer le produit scalaire $\left(\mathbf{Q}^*_j \,.\, \mathbf{Q}^*_k\right)$ sous la forme :

$$\left(\mathbf{Q}^*_j \,.\, \mathbf{Q}^*_k\right) = \begin{cases} r^2 - \dfrac{L_p^2}{2} & \text{si } j \neq k \text{ et } p = \mathrm{comp}(j,k) \\ r^2 & \text{si } j = k \end{cases} \tag{3.22}$$

où L_p est la longueur de l'arête déformée joignant les sommets $\mathbf{Q}_j$ et $\mathbf{Q}_k$.
Par conséquent, C peut s'écrire :

$$\begin{aligned} C &= \sum_j r^2\,(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_j) + \sum_{\substack{j,k\\ j\neq k}} \left(r^2 - \frac{L_p^2}{2}\right)(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k) \\ &= r^2 \sum_{j,k} (\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k) - \frac{1}{2}\sum_{\substack{j,k\\ j\neq k}} L_p^2\,(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k) \\ &= -\frac{1}{2}\sum_{\substack{j,k\\ j\neq k}} L_p^2\,(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k) \end{aligned} \tag{3.23}$$

1. Dans le cas des triangles, les indices ont pour valeurs 0, 1 et 2, ce qui signifie que toutes les opérations sur ces indices se font modulo 3. Afin d'alléger les notations, on considérera la notation "$j+1$" $= (j+1)$ modulo(3)

3.4.1.2 L'énergie potentielle de déformation

L'expression de l'énergie potentielle de déformation repose sur les calculs de la trace de C et de la trace de C^2 :

$$\begin{aligned} W &= \frac{\lambda}{2}(tr\ E)^2 + \mu tr\ E^2 && (3.24)\\ tr\ E &= \frac{1}{2}\boxed{tr\ C} - 1 && (3.25)\\ tr\ E^2 &= \frac{1}{4}\boxed{tr\ C^2} - tr\ E - \frac{1}{2} && (3.26)\end{aligned}$$

En utilisant les propriétés du produit tensoriel (voir section 1.2), on peut exprimer la trace de C de la manière suivante :

$$tr\ C = -\frac{1}{2}\sum_{\substack{j,k\\ j\neq k}} L_p^2\,(\boldsymbol{\alpha}_j\ .\ \boldsymbol{\alpha}_k)\ . \tag{3.27}$$

À partir des relations classiques dans le triangle T^0 (loi des cosinus), on peut montrer que :

$$\overrightarrow{\mathbf{P}_p\mathbf{P}_q}\ .\ \overrightarrow{\mathbf{P}_p\mathbf{P}_r} = \begin{cases} l_j\ l_k\ cos(\theta_p) = \frac{1}{2}\left(l_j^2 + l_k^2 - l_p^2\right) = \mathcal{K}_p & \text{si } q \neq r,\ j = comp(p,q) \\ & \text{et } k = comp(p,r) \\ l_j^2 & \text{si } q = r,\ j = comp(p,q) \end{cases} \tag{3.28}$$

On obtient des relations équivalentes avec les normales aux cotés du triangles :

$$(\boldsymbol{\alpha}_j\ .\ \boldsymbol{\alpha}_k) = \begin{cases} -\frac{l_j\ l_k}{4\mathcal{A}^2}\,cos(\theta_p) = -\frac{1}{4\mathcal{A}^2}\mathcal{K}_p & \text{si } j \neq k \text{ et } p = comp(j,k), \\ \frac{l_j^2}{4\mathcal{A}^2} & \text{si } j = k \end{cases} \tag{3.29}$$

D'où l'on déduit naturellement que :

$$tr\ C = \frac{1}{4\mathcal{A}^2}\sum_p L_p^2\,\mathcal{K}_p\ . \tag{3.30}$$

On peut vérifier assez facilement à partir de la formule de Héron que lorsque le triangle n'est pas déformé ($L_p = l_p$, $p = 0..2$), la trace de C vaut bien 2 :

$$\sum_p l_p^2\,\mathcal{K}_p = 8\mathcal{A}^2 \implies tr\ C = \frac{8\mathcal{A}^2}{4\mathcal{A}^2} = 2\ . \tag{3.31}$$

Ce qui permet de déduire l'expression de $tr\ E$ (équation 3.25) :

$$\boxed{tr\ E = \frac{1}{8\mathcal{A}^2}\sum_p\left(L_p^2 - l_p^2\right)\mathcal{K}_p = \frac{1}{8\mathcal{A}^2}\left[\left(L_0^2 - l_0^2\right)\mathcal{K}_0 + \left(L_1^2 - l_1^2\right)\mathcal{K}_1 + \left(L_2^2 - l_2^2\right)\mathcal{K}_2\right]} \tag{3.32}$$

On obtient donc la première partie de l'énergie élastique en élevant ce terme au carré. Nous n'allons pas développer ce carré puisqu'il est destiné à être dérivé pour calculer la force élastique. On vérifie bien que ce terme ne dépend que des variations de longueur des arêtes du triangle et de sa forme au repos (représentée par les grandeurs $\mathcal{K}_i$, $i = 0..2$).

En élevant l'équation 3.23 au carré et en prenant la trace du résultat, on trouve l'expression de la trace de C^2 :

$$tr\ C^2 = \frac{1}{4} \sum_{\substack{j,k \\ j\neq k}} \sum_{\substack{m,n \\ m\neq n}} L_p^2\, L_q^2\, (\boldsymbol{\alpha}_j \,.\, \boldsymbol{\alpha}_n)\, (\boldsymbol{\alpha}_k \,.\, \boldsymbol{\alpha}_m) \tag{3.33}$$

À ce stade, il faut remarquer que les quatre indices (j,k,m,n) ne peuvent prendre que les valeurs (0,1,2), et que par conséquent il y aura toujours au moins deux de ces indices qui seront égaux. Nous utilisons cette propriété pour scinder l'expression de $tr\ C^2$ en deux parties distinctes correspondant aux cas où ($j = m$ ou $k = n$) et ($j = n$ ou $k = m$) (n.b. : le "ou" est non exclusif) :

$$\begin{aligned} tr\ C^2 \;&=\; \frac{1}{4} \sum_{\substack{j,k \\ j\neq k}} L_{jk}^4\, (\boldsymbol{\alpha}_j \,.\, \boldsymbol{\alpha}_k)^2 + \frac{1}{2} \sum_{\substack{j,k \\ j\neq k}} \sum_{\substack{m \\ m\neq j \\ m\neq k}} L_{jk}^2\, L_{jm}^2\, (\boldsymbol{\alpha}_j \,.\, \boldsymbol{\alpha}_k)\, (\boldsymbol{\alpha}_j \,.\, \boldsymbol{\alpha}_m) & (3.34) \\ &+\; \frac{1}{4} \sum_{\substack{j,k \\ j\neq k}} L_{jk}^4\, \boldsymbol{\alpha}_j^2\, \boldsymbol{\alpha}_k^2 + \frac{1}{2} \sum_{\substack{j,k,m \\ j\neq k}} \sum_{\substack{m \\ m\neq j \\ m\neq k}} L_{jk}^2\, L_{jm}^2\, \boldsymbol{\alpha}_j^2\, (\boldsymbol{\alpha}_k \,.\, \boldsymbol{\alpha}_m) & (3.35) \end{aligned}$$

Considérons tout d'abord la première ligne de cette équation (3.34). Le premier terme parcourt deux fois chaque arête $(j,k) = p$ du triangle, tandis que le second terme parcourt deux fois chaque couple d'arêtes $((j,k)(j,m)) = (p,q)$. À l'aide de l'équation 3.29, on peut alors mettre cette première partie de l'équation sous la forme :

$$\frac{1}{32\mathcal{A}^4} \sum_{p} L_p^4\, \mathcal{K}_p^4 + \frac{1}{16\mathcal{A}^4} \sum_{\substack{p,q \\ p<q}} L_p^2\, L_q^2 \mathcal{K}_p\, \mathcal{K}_q = \boxed{\frac{1}{2}(tr\ C)^2}$$

On peut raisonner de la même manière pour les termes de la seconde ligne (3.35), et en se rappelant que $\boldsymbol{\alpha}_i^2 = l_i^2$, on arrive à la forme suivante :

$$\begin{aligned} tr\ C^2 = \frac{1}{2}(tr\ C)^2 \;&+\; \frac{1}{32\mathcal{A}^4} \left(L_0^4\, l_1^2 l_2^2 + L_1^4\, l_0^2 l_2^2 + L_2^4\, l_0^2 l_1^2\right) & (3.36) \\ &-\; \frac{1}{16\mathcal{A}^4} \left(L_0^2 L_1^2\, l_2^2 \mathcal{K}_2 + L_0^2 L_2^2\, l_1^2 \mathcal{K}_1 + L_1^2 L_2^2\, l_0^2 \mathcal{K}_0\right) & (3.37) \end{aligned}$$

À partir des équations 3.25 et 3.26, on peut finalement déduire que :

$$\begin{aligned} tr\ E^2 \;&=\; \frac{1}{2}(tr\ E)^2 \\ &+\; \frac{1}{128\mathcal{A}^4} \left[\left(L_0^2 - l_0^2\right)^2 l_1^2 l_2^2 + \left(L_1^2 - l_1^2\right)^2 l_0^2 l_2^2 + \left(L_2^2 - l_2^2\right)^2 l_0^2 l_1^2\right] & (3.38) \\ &-\; \frac{1}{64\mathcal{A}^4} \left[\left(L_0^2 - l_0^2\right)\left(L_1^2 - l_1^2\right) l_2^2 \mathcal{K}_2 + \left(L_0^2 - l_0^2\right)\left(L_2^2 - l_2^2\right) l_1^2 \mathcal{K}_1 + \left(L_1^2 - l_1^2\right)\left(L_2^2 - l_2^2\right) l_0^2 \mathcal{K}_0\right] \end{aligned}$$

Finalement, l'énergie potentielle élastique de déformation du triangle est donnée par la formule suivante :

$$\begin{aligned} W \;=\; & \frac{1}{128\mathcal{A}^4}\Bigg[(\lambda+\mu)\left(\sum_p \left(L_p^2-l_p^2\right)\mathcal{K}_p\right)^2 \\ & + \mu\left(\sum_p \left(L_p^2-l_p^2\right)^2\; l_j^2 l_k^2 - 2\sum_{\substack{j,k\\ j\neq k}} \left(L_j^2-l_j^2\right)\left(L_k^2-l_k^2\right)\; l_p^2\mathcal{K}_p\right)\Bigg] \qquad (3.39) \\ & p = comp(j,k) \end{aligned}$$

3.4.1.3 Expression des forces élastiques

On trouve la force élastique appliquée par le triangle sur un de ses sommets en dérivant l'énergie élastique par rapport à la position de ce sommet. Prenons par exemple le sommet $\mathbf{P}_0$, qui a pour position déformée $\mathbf{Q}_0$. Il faut alors dériver l'équation 3.39 par rapport à $\mathbf{Q}_0$. Or les termes dépendant de $\mathbf{Q}_0$ sont les carrés des longueurs des arêtes partant de ce sommet. On commence donc par dériver les termes L_p^2 :

$$\frac{\partial L_p^2}{\partial \mathbf{Q}_0} = \frac{\partial \|\overrightarrow{\mathbf{Q}_j\mathbf{Q}_k}\|^2}{\partial \mathbf{Q}_0} = \begin{cases} 0 & \text{si } j\neq 0 \text{ et } k\neq 0 \\ 2\overrightarrow{\mathbf{Q}_j\mathbf{Q}_0} & \text{si } k=0 \\ 2\overrightarrow{\mathbf{Q}_k\mathbf{Q}_0} & \text{si } j=0 \end{cases} \qquad (3.40)$$

Ceci permet de dériver l'équation 3.39 et ainsi d'obtenir l'expression suivante :

$$\begin{aligned} \mathbf{F}_0^T = \frac{\partial W}{\partial \mathbf{Q}_0} \;=\; & \frac{1}{32\mathcal{A}^4}\Bigg[(\lambda+\mu)\left(\sum_p \left(L_p^2-l_p^2\right)\mathcal{K}_p\right)\left(\overrightarrow{\mathbf{Q}_2\mathbf{Q}_0}\;\mathcal{K}_1 + \overrightarrow{\mathbf{Q}_1\mathbf{Q}_0}\;\mathcal{K}_2\right) \qquad (3.41) \\ + \;& \mu\Bigg(\left(L_1^2-l_1^2\right) l_0^2\left(l_2^2\;\overrightarrow{\mathbf{Q}_2\mathbf{Q}_0} - \mathcal{K}_0\overrightarrow{\mathbf{Q}_1\mathbf{Q}_0}\right) + \left(L_2^2-l_2^2\right) l_0^2\left(l_1^2\;\overrightarrow{\mathbf{Q}_1\mathbf{Q}_0} - \mathcal{K}_0\overrightarrow{\mathbf{Q}_2\mathbf{Q}_0}\right) \\ & -\left(L_0^2-l_0^2\right)\left(l_1^2\;\mathcal{K}_1\;\overrightarrow{\mathbf{Q}_1\mathbf{Q}_0} + l_2^2\;\mathcal{K}_2\;\overrightarrow{\mathbf{Q}_2\mathbf{Q}_0}\right)\Bigg)\Bigg] \end{aligned}$$

Nous avons donc obtenu l'équation de la force élastique exercée par le triangle sur chacun de ses sommets. Cette force s'exprime bien uniquement en fonction des variations de longueur des arêtes du triangle ($L_i^2 - l_i^2$), des caractéristiques biomécaniques du matériau (λ, μ) et de la forme du triangle au repos ($\mathcal{K}_i$, $i = 0..2$). Cette force contient implicitement la propriété d'invariance par rotation puisque tous ses termes restent nuls tant que les longueurs des arêtes du triangle de changent pas, c'est-à-dire tant que le triangle ne se déforme pas.

3.4.1.4 Interprétation des forces

Essayons maintenant de trouver une interprétation aux différents termes de cette force. Commençons par le terme en $(\lambda + \mu)$. La première partie est une valeur scalaire, qui peut s'écrire sous la forme :

$$\sum_p (L_p^2 - l_p^2)\,\mathcal{K}_p = \Big(\sum_p L_p^2\,\mathcal{K}_p\Big) - \mathcal{A}^2$$

Elle s'apparente donc à une variation de l'aire du triangle. La seconde partie donne la direction de la force. On peut tout d'abord remarquer que lorsque le triangle a sa forme de repos, cette force est perpendiculaire à l'arête opposée :

$$\overrightarrow{n}_0 \,.\, \overrightarrow{\mathbf{P}_1\mathbf{P}_2} = -\mathcal{K}_1\overrightarrow{\mathbf{P}_2\mathbf{P}_0} \,.\, \overrightarrow{\mathbf{P}_2\mathbf{P}_1} + \mathcal{K}_2\overrightarrow{\mathbf{P}_1\mathbf{P}_0} \,.\, \overrightarrow{\mathbf{P}_1\mathbf{P}_2} = -\mathcal{K}_1\,\mathcal{K}_2 + \mathcal{K}_2\,\mathcal{K}_1 = 0.$$

On introduit donc $\mathbf{h}_0$, le pied de la hauteur issue de $\mathbf{P}_0$, et on exprime sa position sur la droite $(\mathbf{P}_1,\mathbf{P}_2)$ par :

$$\overrightarrow{\mathbf{P}_1\mathbf{h}_0} = \frac{\mathcal{K}_1}{l_0^2}\,\overrightarrow{\mathbf{P}_1\mathbf{P}_2} \qquad \text{et} \qquad \overrightarrow{\mathbf{P}_2\mathbf{h}_0} = \frac{\mathcal{K}_2}{l_0^2}\,\overrightarrow{\mathbf{P}_2\mathbf{P}_1} \tag{3.42}$$

Une fois le triangle déformé, ce point devient $\mathbf{H}_0$ et on a (voir figure 3.12) :

$$\overrightarrow{\mathbf{Q}_1\mathbf{H}_0} = \frac{\mathcal{K}_1}{l_0^2}\,\overrightarrow{\mathbf{Q}_1\mathbf{Q}_2} \qquad \text{et} \qquad \overrightarrow{\mathbf{Q}_2\mathbf{H}_0} = \frac{\mathcal{K}_2}{l_0^2}\,\overrightarrow{\mathbf{Q}_2\mathbf{Q}_1} \tag{3.43}$$

On peut alors montrer que la force est dirigée le long de la droite $(\mathbf{Q}_0,\mathbf{H}_0)$, puisque :

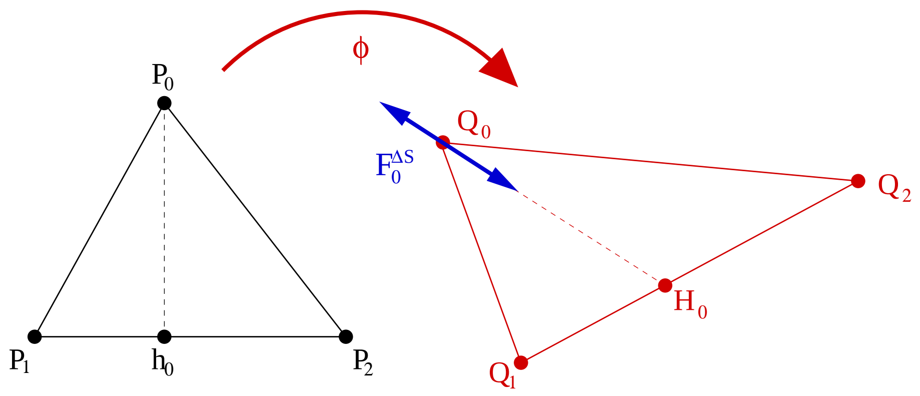


FIG. 3.12 – *Orientation du premier terme de la force élastique (équation 3.41).*

$$\begin{aligned}
\overrightarrow{\mathbf{Q}_2\mathbf{Q}_0}\,\mathcal{K}_1 + \overrightarrow{\mathbf{Q}_1\mathbf{Q}_0}\,\mathcal{K}_2 &= (\mathcal{K}_1+\mathcal{K}_2)\overrightarrow{\mathbf{H}_0\mathbf{Q}_0} + \big(\overrightarrow{\mathbf{Q}_1\mathbf{H}_0}\mathcal{K}_2 + \overrightarrow{\mathbf{Q}_2\mathbf{H}_0}\mathcal{K}_1\big)\\
&= l_0^2\,\overrightarrow{\mathbf{H}_0\mathbf{Q}_0} + \Big(\frac{\mathcal{K}_1\mathcal{K}_2}{l_0^2} - \frac{\mathcal{K}_1\mathcal{K}_2}{l_0^2}\Big)\overrightarrow{\mathbf{Q}_1\mathbf{Q}_2}\\
&= l_0^2\,\overrightarrow{\mathbf{H}_0\mathbf{Q}_0}
\end{aligned}$$

On appelle donc cette force qui est reliée à la variation d'aire du triangle $\mathbf{F}_0^{\Delta S}$ et on l'exprime sous la forme :

$$\boxed{\mathbf{F}_0^{\Delta S} = \Big[\Big(\sum_p L_p^2\,\mathcal{K}_p\Big) - \mathcal{A}^2\Big]\,l_0^2\,\overrightarrow{\mathbf{H}_0\mathbf{Q}_0}} \tag{3.44}$$

Le terme en μ est constitué de trois forces, chacune étant reliée à la variation de longueur d'une des arêtes du triangle. On les nommera donc logiquement :

$$\begin{cases} \mathbf{F}_0^{\Delta l_0} = -\left(L_0^2 - l_0^2\right)\left(l_1^2\,\mathcal{K}_1\,\overrightarrow{\mathbf{Q}_1\mathbf{Q}_0} + l_2^2\,\mathcal{K}_2\,\overrightarrow{\mathbf{Q}_2\mathbf{Q}_0}\right) \\ \mathbf{F}_0^{\Delta l_1} = \left(L_1^2 - l_1^2\right)\,l_0^2\left(l_2^2\,\overrightarrow{\mathbf{Q}_2\mathbf{Q}_0} - \mathcal{K}_0\overrightarrow{\mathbf{Q}_1\mathbf{Q}_0}\right) \\ \mathbf{F}_0^{\Delta l_2} = \left(L_2^2 - l_2^2\right)\,l_0^2\left(l_1^2\,\overrightarrow{\mathbf{Q}_1\mathbf{Q}_0} - \mathcal{K}_0\overrightarrow{\mathbf{Q}_2\mathbf{Q}_0}\right) \end{cases}$$

Commençons par les deux dernières, qui sont parfaitement symétriques. De la même manière que précédemment, on remarque que lorsque le triangle est au repos, la direction de la force $\mathbf{F}_0^{\Delta l_1}$ est perpendiculaire au coté $[\mathbf{P}_0,\mathbf{P}_2]$. On construit alors $\mathbf{h}_2$, le pied de la hauteur issue de $\mathbf{P}_2$, et on exprime sa position sur la droite $(\mathbf{P}_0,\mathbf{P}_2)$ par des formules équivalentes aux équations 3.42 et 3.43. On montre alors que lorsque le triangle se déforme, la force $\mathbf{F}_0^{\Delta l_1}$ a pour direction la droite $[\mathbf{Q}_2,\mathbf{H}_2]$, où $\mathbf{H}_2$ est la position déformée du point $\mathbf{h}_2$ (voir figure 3.13) :

$$\begin{aligned} l_2^2\,\overrightarrow{\mathbf{Q}_2\mathbf{Q}_0} - \mathcal{K}_0\overrightarrow{\mathbf{Q}_1\mathbf{Q}_0} &= l_2^2\,\overrightarrow{\mathbf{Q}_2\mathbf{H}_2} + l_2^2\,\overrightarrow{\mathbf{H}_2\mathbf{Q}_0} - \mathcal{K}_0\overrightarrow{\mathbf{Q}_1\mathbf{Q}_0} \\ &= l_2^2\,\overrightarrow{\mathbf{Q}_2\mathbf{H}_2} + \left(l_2^2\,\frac{\mathcal{K}_0}{l_2^2} - \mathcal{K}_0\right)\overrightarrow{\mathbf{Q}_1\mathbf{Q}_0} \\ &= l_2^2\,\overrightarrow{\mathbf{Q}_2\mathbf{H}_2} \end{aligned}$$

On peut donc exprimer les deux forces $\mathbf{F}_0^{\Delta l_1}$ et $\mathbf{F}_0^{\Delta l_2}$ de la manière suivante :

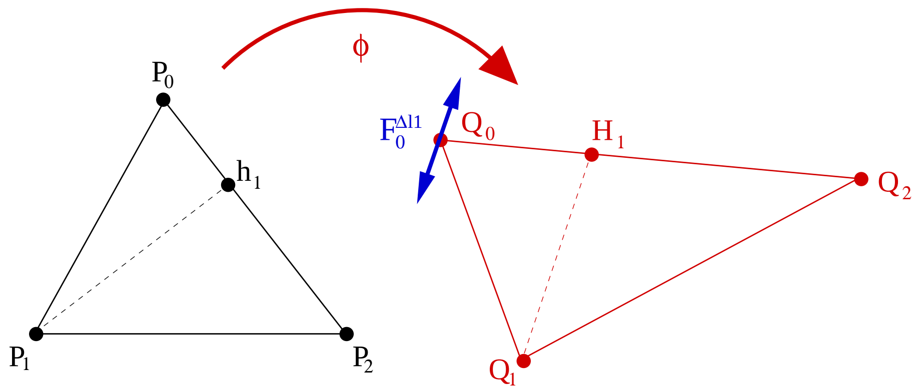


FIG. 3.13 – *Orientation de la force* $\mathbf{F}_0^{\Delta l_1}$.

$$\boxed{\mathbf{F}_0^{\Delta l_1} = \left(L_1^2 - l_1^2\right)\,l_0^2\,l_2^2\,\overrightarrow{\mathbf{Q}_2\mathbf{H}_2}} \tag{3.45}$$

$$\boxed{\mathbf{F}_0^{\Delta l_2} = \left(L_2^2 - l_2^2\right)\,l_0^2\,l_1^2\,\overrightarrow{\mathbf{Q}_1\mathbf{H}_1}} \tag{3.46}$$

Pour la force liée à la variation de l_0, on suit le même genre de raisonnement : trouver un point $\mathbf{g}$ du triangle au repos tel que la force $\mathbf{F}_0^{\Delta l_0}$ ait pour direction la droite $(\mathbf{P}_0,\mathbf{g})$. On introduit donc le point $\mathbf{g}$ dans l'expression de la force :

$$l_1^2\,\mathcal{K}_1\,\overrightarrow{\mathbf{P}_1\mathbf{P}_0} + l_2^2\,\mathcal{K}_2\,\overrightarrow{\mathbf{P}_2\mathbf{P}_0} = \left(l_1^2\,\mathcal{K}_1 + l_2^2\,\mathcal{K}_2\right)\,\overrightarrow{\mathbf{g}\mathbf{P}_0} + \left(l_1^2\,\mathcal{K}_1\,\overrightarrow{\mathbf{P}_1\mathbf{g}} + l_2^2\,\mathcal{K}_2\,\overrightarrow{\mathbf{P}_2\mathbf{g}}\right) \tag{3.47}$$

Le dernier terme de l'équation précédente incite à définir $\mathbf{g}$ comme le barycentre des sommets du triangle affectés des cœfficients $\big((\mathbf{P}_0,\, l_0^2\, \mathcal{K}_0),(\mathbf{P}_1,\, l_1^2\, \mathcal{K}_1),(\mathbf{P}_2,\, l_2^2\, \mathcal{K}_2)\big)$.
Considérons le centre $\mathbf{c}$ du cercle circonscrit. Les coordonnées barycentriques de ce point dans le triangle sont les rapports $\mathcal{A}_i/\mathcal{A}$, où $\mathcal{A}_i$ est l'aire du sous-triangle $(\mathbf{c},\, \mathbf{P}_{i+1},\, \mathbf{P}_{i+2})$ et $\mathcal{A}$ est l'aire du triangle :

$$\begin{cases} \mathbf{P}_1\, \widehat{\mathbf{c}}\, \mathbf{I}_0 = \mathbf{P}_1 \widehat{\mathbf{P}_0} \mathbf{P}_2 = \theta_0 \text{ (théorème de l'angle au centre)} \\ \mathcal{A}_0 = \dfrac{1}{2}\, l_0\, d_0 \\ d_0 = \dfrac{1}{2}\, l_0\, cot(\theta_0) \\ cot(\theta_0) = \dfrac{-l_0^2 + l_1^2 + l_2^2}{4\mathcal{A}} = \dfrac{\mathcal{K}_0}{2\mathcal{A}} \end{cases}$$

$$\Rightarrow \frac{\mathcal{A}_0}{\mathcal{A}} = \frac{l_0\, d_0}{2\mathcal{A}} = \frac{l_0^2\, cot(\theta_0)}{4\mathcal{A}} = \frac{l_0^2\, \mathcal{K}_0}{8\mathcal{A}^2}$$

et on vérifie bien que :

$$\frac{l_0^2\, \mathcal{K}_0}{8\mathcal{A}^2} + \frac{l_1^2\, \mathcal{K}_1}{8\mathcal{A}^2} + \frac{l_2^2\, \mathcal{K}_2}{8\mathcal{A}^2} = 1$$

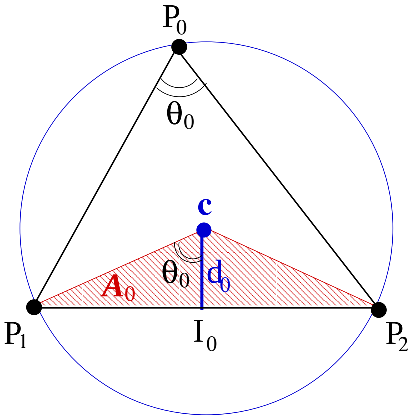


FIG. 3.14 – *Calcul des coordonnées barycentriques du centre du cercle circonscrit.*

Le point $\mathbf{g}$ que l'on cherchait est donc bien le centre du cercle circonscrit. Dans le triangle déformé, $\mathbf{C}$ est l'image de $\mathbf{c}$ par $\mathbf{\Phi}$ et vérifie donc l'équation :

$$l_0^2\, \mathcal{K}_0\, \overrightarrow{\mathbf{Q}_0\mathbf{C}} + l_1^2\, \mathcal{K}_1\, \overrightarrow{\mathbf{Q}_1\mathbf{C}} + l_2^2\, \mathcal{K}_2\, \overrightarrow{\mathbf{Q}_2\mathbf{C}} = \overrightarrow{0}$$

En reportant cette équation dans la formule 3.47 appliquée au triangle déformé, on obtient :

$$l_1^2\, \mathcal{K}_1\, \overrightarrow{\mathbf{Q}_1\mathbf{Q}_0} + l_2^2\, \mathcal{K}_2\, \overrightarrow{\mathbf{Q}_2\mathbf{Q}_0} = \big(l_0^2\, \mathcal{K}_0 + l_1^2\, \mathcal{K}_1 + l_2^2\, \mathcal{K}_2\big)\, \overrightarrow{\mathbf{C}\mathbf{Q}_0}$$

Ce qui permet de donner l'expression de la force $\mathbf{F}_0^{\Delta l_0}$ (figure 3.15) :

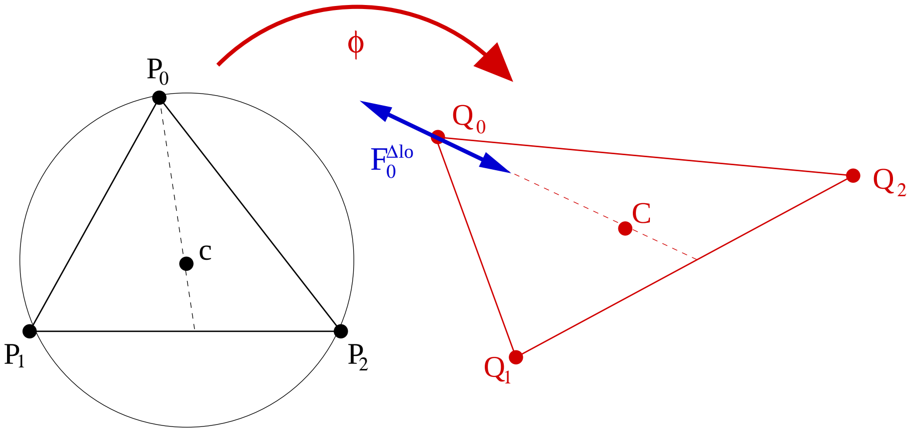


FIG. 3.15 – *Orientation de la force* $\mathbf{F}_0^{\Delta l_0}$.

$$\boxed{\mathbf{F}_0^{\Delta l_2} = 8\big(L_0^2 - l_0^2\big)\, \mathcal{A}^2\, \overrightarrow{\mathbf{Q}_0\mathbf{C}}} \tag{3.48}$$

Finalement, en regroupant toutes ces équations, on peut écrire la force élastique exercée par le triangle sur le sommet $\mathbf{Q}_0$ sous la forme :

$$\mathbf{F}_0^T = \frac{1}{32\mathcal{A}^4}\Bigg[(\lambda+\mu)\left[\Big(\sum_p L_p^2\,\mathcal{K}_p\Big) - \mathcal{A}^2\right] l_0^2\;\overrightarrow{\mathbf{H}_0\mathbf{Q}_0} \tag{3.49}$$
$$+\mu\left[8\big(L_0^2-l_0^2\big)\,\mathcal{A}^2\;\overrightarrow{\mathbf{Q}_0\mathbf{C}} + \big(L_1^2-l_1^2\big)\,l_0^2\,l_2^2\;\overrightarrow{\mathbf{Q}_2\mathbf{H}_2} + \big(L_2^2-l_2^2\big)\,l_0^2\,l_1^2\;\overrightarrow{\mathbf{Q}_1\mathbf{H}_1}\right]\Bigg]$$

Cette expression de la force élastique n'est pas vraiment plus simple que l'autre, mais elle permet de comprendre comment le comportement élastique du triangle se traduit du point de vue géométrique.

3.4.1.5 Mise en œuvre et applications

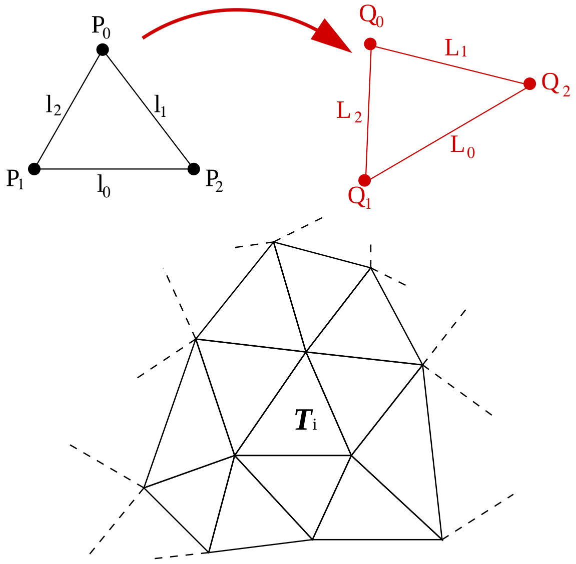
La formulation que nous venons de détailler permet d'exprimer l'énergie de déformation et les forces élastiques pour un élément de surface triangulaire ayant le comportement d'un matériau de St Venant-Kirchhoff. Nous sommes donc en mesure de construire un modèle déformable surfacique à partir de la triangulation de la frontière externe d'un objet. La mise en œuvre de ce nouveau modèle déformable est assez proche de ce que nous avons fait avec le modèle masse-tenseur. Lors de la création du maillage triangulaire, certaines informations géométriques sont stockées sur les arêtes et les sommets du maillage. Ces informations caractérisent la forme du modèle au repos. Ensuite, lors du calcul de la déformation, chaque triangle du maillage est parcouru afin d'évaluer les forces qu'il exerce sur ses trois sommets. Une fois les forces accumulées sur les sommets, leurs nouvelles positions sont déduites de l'intégration des équations du mouvement :

Initialisation :

- Pour chaque arête, calculer le carré de sa longueur au repos.
- Pour chaque triangle, calculer les 3 termes $\mathcal{K}_j$ (équation 3.28), et son aire au repos.

Calcul de la déformation :

- Pour chaque triangle $\mathcal{T}_i$ du maillage, calculer la force élastique exercée sur chacun des sommets (équation 3.41).
- Pour chaque sommet, calculer la nouvelle position par intégration du mouvement (équation 2.32 du paragraphe 2.4.2.3).



Nous avons tout d'abord vérifié que les forces que nous calculons minimisent bien l'énergie élastique du modèle. Pour cela, nous avons déformé un modèle, que nous avons ensuite laissé évoluer en calculant, à chaque pas de temps, l'énergie globale (équation 3.39). La figure 3.16 montre que les forces élastiques déduites de la déformation font effectivement décroître l'énergie du modèle.

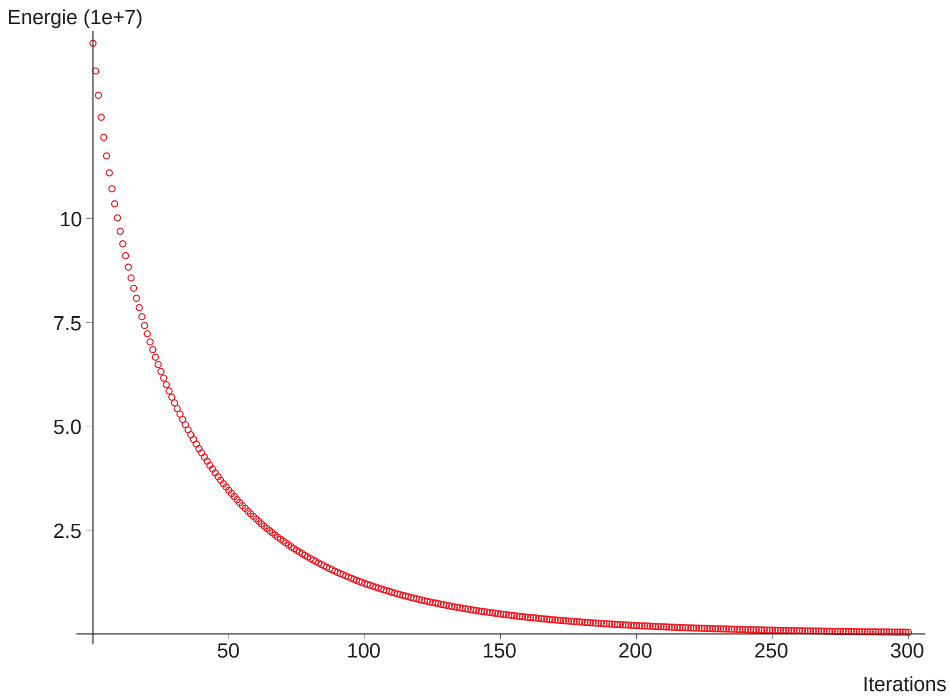


FIG. 3.16 – *Minimisation de l'énergie de déformation.*

La figure 3.17 donne un exemple du comportement obtenu avec ce modèle déformable surfacique. Pour avoir un ordre d'idée des performances de ce modèle, nous l'avons comparé avec un modèle masse-ressort. Avec notre mise en œuvre actuelle, le modèle élastique non-linéaire s'avère être 3.5 fois moins rapide que le masse-ressort, ce qui nous semble être un bon résultat si l'on considère le réalisme biomécanique que ce nouveau modèle apporte.

Nous envisageons d'utiliser ce nouveau modèle déformable surfacique pour deux applications. La première est la modélisation d'organes creux, comme par exemple la vésicule biliaire ou l'estomac. Dans ce cas nous voulons lui adjoindre une pression interne qui pénalisera la variation globale de volume de l'organe. Nous pensons ainsi simuler le comportement d'un organe creux rempli de liquide. La deuxième application possible est la modélisation de la capsule de Glisson, ou de toute autre peau qui rend la surface de certains organes plus rigide. Ce modèle surfacique peut, dans ce cas, être couplé avec n'importe quel modèle déformable volumique comportant une surface triangulée.

3.4.2 Élasticité 3D : tétraèdre hyperélastique

Dans cette partie, nous étendons au cas volumique la formulation "simplexe hyperélastique" de l'élasticité de St Venant-Kirchhoff. Même si les raisonnements se font maintenant sur des tétraèdres (au lieu de triangles), la plupart des équations demeurent très proches du cas 2D.

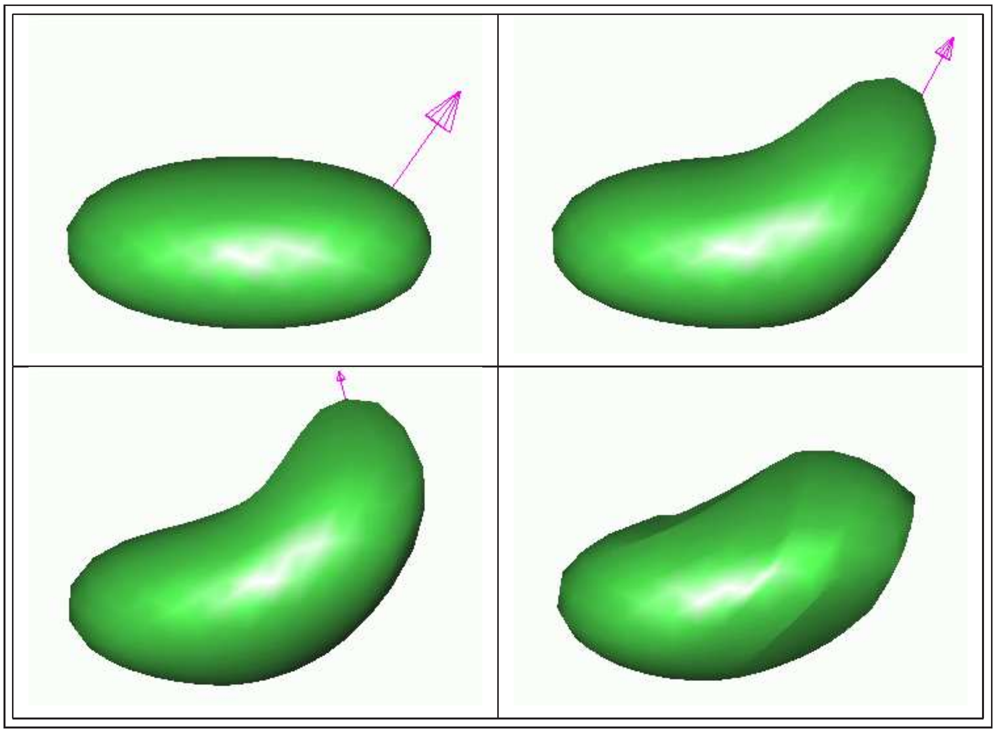


FIG. 3.17 – *Déformation d'un modèle surfacique sous l'influence de contraintes externes.*

3.4.2.1 Le tenseur de déformation

On considère un tétraèdre $T^0 = (\mathbf{P}_0,\mathbf{P}_1,\mathbf{P}_2,\mathbf{P}_3)$ de l'espace $\mathbb{R}^3$, sur lequel on définit les longueurs des arêtes $(l_{01},l_{02},l_{03},l_{12},l_{13},l_{23})$ et les angles dièdres entre les faces $(\theta_{01},\theta_{02},\theta_{03},\theta_{12},\theta_{13},\theta_{23})$. On applique à ce tétraèdre T^0 une déformation $\mathbf{\Phi}$, pour obtenir le tétraèdre déformé $T = (\mathbf{Q}_0,\mathbf{Q}_1,\mathbf{Q}_2,\mathbf{Q}_3)$ dont les longueurs d'arêtes sont $(L_{01}, L_{02}, L_{03}, L_{12}, L_{13}, L_{23})$.
En considérant le tétraèdre T^0 comme un élément fini de type P_1, on peut exprimer le gradient de la déformée, ainsi que le tenseur de déformation de Cauchy-Green C, de la manière suivante :

$$\nabla\mathbf{\Phi} = \sum_{j=0}^{3} \boldsymbol{\alpha}_j \otimes \mathbf{Q}_j \qquad \boldsymbol{\alpha}_j = \frac{\mathcal{A}_j}{3\mathcal{V}}\mathbf{n}_j \tag{3.50}$$

$$C = \sum_{j,k} (\mathbf{Q}_j \,.\, \mathbf{Q}_k)\,(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k) \tag{3.51}$$

où $\mathcal{A}_j$ est l'aire de la face opposée au sommet $\mathbf{P}_j$, $\mathbf{n}_j$ est la normale à cette face et $\mathcal{V}$ est le volume du tétraèdre (voir paragraphe 1.3 pour plus de détails).

En plaçant le centre du repère au centre de la sphère circonscrite au tétraèdre T, on peut exprimer le produit scalaire des vecteurs position de deux sommets du tétraèdre en fonction de la longueur de l'arête qui les joint et du rayon r de la sphère, ce qui permet d'écrire le tenseur de déformation sous la forme :

$$(\mathbf{Q}_j \ . \ \mathbf{Q}_k) = \begin{cases} r^2 - \dfrac{L_{jk}^2}{2} & \text{si } j \neq k \\ r^2 & \text{si } j = k \end{cases} \qquad \Longrightarrow \qquad \boxed{C = -\frac{1}{2} \sum_{\substack{j,k \\ j \neq k}} L_p^2 \, (\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k)} \tag{3.52}$$

3.4.2.2 L'énergie potentielle de déformation

L'énergie élastique de déformation s'exprime de la même manière que dans le cas 2D en fontion du tenseur de déformation de Green-St Venant E. Par contre les relations entre trE et trC sont légèrement différentes :

$$\begin{aligned} W &= \frac{\lambda}{2} (tr \ E)^2 + \mu tr \ E^2 \\ tr \ E &= \frac{1}{2} \boxed{tr \ C} - \frac{3}{2} \\ tr \ E^2 &= \frac{1}{4} \boxed{tr \ C^2} - tr \ E - \frac{3}{4} \end{aligned} \tag{3.53}$$

Considérons tout d'abord la trace de C. Son expression fait apparaître le terme $(\boldsymbol{\alpha}_j \ . \ \boldsymbol{\alpha}_k)$. À partir de la loi des cosinus pour les angles dièdres, on exprime ce terme en fonction des longueurs des arêtes du tétraèdre :

$$\begin{aligned} (\boldsymbol{\alpha}_j \ . \ \boldsymbol{\alpha}_k) &= \begin{cases} -\dfrac{1}{9\mathcal{V}^2} \mathcal{K}_{jk} & \text{si } j \neq k \\ \dfrac{1}{9\mathcal{V}^2} \mathcal{A}_j^2 & \text{si } j = k \end{cases} \\ \text{avec } \ \mathcal{K}_{jk} &= \frac{l_{lm}^2 \left(l_{jl}^2 + l_{kl}^2 + l_{jm}^2 + l_{km}^2 - 2 \, l_{jk}^2 - l_{lm}^2 \right) - \left(l_{jl}^2 - l_{jm}^2 \right) \left(l_{kl}^2 - l_{km}^2 \right)}{16} \end{aligned} \tag{3.54}$$

On peut alors exprimer la trace de C, puis la trace de E, de manière équivalente au cas 2D :

$$tr \ C = \frac{1}{9\mathcal{V}^2} \sum_{\text{Arêtes(j,k)}} L_{jk}^2 \, \mathcal{K}_{jk} \tag{3.55}$$

$$\boxed{tr \ E = \frac{1}{18\mathcal{V}^2} \sum_{\text{Arêtes(j,k)}} \left(L_{jk}^2 - l_{jk}^2 \right) \mathcal{K}_{jk}} \tag{3.56}$$

La trace de C^2 reprend, elle aussi, exactement la même forme que pour le cas 2D :

$$tr \ C^2 = \frac{1}{4} \sum_{\substack{j,k \\ j \neq k}} \sum_{\substack{m,n \\ m \neq n}} L_{jk}^2 \, L_{mn}^2 \, (\boldsymbol{\alpha}_j \ . \ \boldsymbol{\alpha}_n) \, (\boldsymbol{\alpha}_k \ . \ \boldsymbol{\alpha}_m) \tag{3.57}$$

Par contre, les indices j, k, m et n prennent cette fois leurs valeurs entre 0 et 3, ce qui signifie qu'il faut aussi envisager le cas où tous les indices sont différents. En décomposant cette somme en fonction des éventuelles égalités entre les indices et en utilisant l'équation 3.54, on arrive à la formule :

$$\begin{aligned} tr\ C^2 = \frac{1}{81\ \mathcal{V}^4}\Bigg[\frac{1}{2} & \sum_{\text{arêtes}(j,k)} \underline{L_{jk}^4}\ \big(\underline{\mathcal{K}_{jk}^2} + \mathcal{A}_j^2\ \mathcal{A}_k^2\big) && \text{6 termes} \\ + & \sum_{\substack{\text{couples d'arêtes} \\ ((j,k),(j,m))}} \underline{L_{jk}^2\ L_{jm}^2}\ \big(\underline{\mathcal{K}_{jk}\ \mathcal{K}_{jm}} - \mathcal{A}_j^2\ \mathcal{K}_{km}\big) && \text{12 termes} \\ + & \sum_{\substack{\text{couples d'arêtes} \\ ((j,k),(m,n))}} L_{jk}^2\ L_{mn}^2\ \big(\mathcal{K}_{jn}\ \mathcal{K}_{km} + \mathcal{K}_{jm}\ \mathcal{K}_{kn}\big)\Bigg] && \text{3 termes} \end{aligned} \tag{3.58}$$

Les termes soulignés dans cette équation (3.58) interviennent dans le développement de $\frac{1}{2}(tr\ C)^2$. On peut donc isoler cette partie de l'équation, en prenant soin de retrancher les termes manquant dans la dernière partie du développement, qui sont encadrés dans l'équation ci-dessous :

$$\begin{aligned} tr\ C^2 = \frac{1}{2}\big(tr\ C\big)^2 + \frac{1}{81\ \mathcal{V}^4}\Bigg[\frac{1}{2} & \sum_{\text{arêtes}(j,k)} L_{jk}^4\ \mathcal{A}_j^2\ \mathcal{A}_k^2 \\ - & \sum_{\substack{\text{couples d'arêtes} \\ ((j,k),(j,m))}} L_{jk}^2\ L_{jm}^2\ \mathcal{A}_j^2\ \mathcal{K}_{km} \\ + & \sum_{\substack{\text{couples d'arêtes} \\ ((j,k),(m,n))}} L_{jk}^2\ L_{mn}^2\ \big(\mathcal{K}_{jn}\ \mathcal{K}_{km} + \mathcal{K}_{jm}\ \mathcal{K}_{kn} \boxed{-\mathcal{K}_{jk}\ \mathcal{K}_{mn}}\big)\Bigg] \end{aligned} \tag{3.59}$$

Si l'on nomme Δ le terme entre crochets, la trace de E^2 se déduit alors des équations 3.53 et 3.59, et s'exprime sous la forme :

$$tr\ E^2 = \frac{1}{2}\big(tr\ E\big)^2 + \frac{1}{2}tr\ E + \frac{3}{8} + \frac{\Delta}{324\ \mathcal{V}^4} \tag{3.60}$$

La suite du raisonnement consiste à faire apparaître, dans Δ, la variation des carrés des longueurs des arêtes du tétraèdre. Pour cela, on remplace L_{jk}^2 par $(L_{jk}^2 - l_{jk}^2) + l_{jk}^2$. En développant, on obtient l'expression suivante de Δ :

$$\begin{aligned} \Delta = & \frac{1}{2} \sum_{\text{arêtes}(j,k)} \Big[\big(L_{jk}^2 - l_{jk}^2\big)^2 + \boxed{2\big(L_{jk}^2 - l_{jk}^2\big)l_{jk}^2} + \underline{l_{jk}^4}\Big]\ \mathcal{A}_j^2\ \mathcal{A}_k^2 \\ - & \sum_{\substack{\text{couples d'arêtes} \\ ((j,k),(j,m))}} \Big[\big(L_{jk}^2 - l_{jk}^2\big)\big(L_{jm}^2 - l_{jm}^2\big) + \boxed{\big(L_{jk}^2 - l_{jk}^2\big)l_{jm}^2} + \boxed{\big(L_{jm}^2 - l_{jm}^2\big)l_{jk}^2} + \underline{l_{jk}^2\ l_{jm}^2}\Big]\ \mathcal{A}_j^2\ \mathcal{K}_{km} \\ + & \sum_{\substack{\text{couples d'arêtes} \\ ((j,k),(m,n))}} \left[\begin{array}{c} \big(L_{jk}^2 - l_{jk}^2\big)\big(L_{mn}^2 - l_{mn}^2\big) + \boxed{\big(L_{jk}^2 - l_{jk}^2\big)l_{mn}^2} \\ + \boxed{\big(L_{mn}^2 - l_{mn}^2\big)l_{jk}^2} + \underline{l_{jk}^2\ l_{mn}^2} \end{array}\right] \big(\mathcal{K}_{jn}\ \mathcal{K}_{km} + \mathcal{K}_{jm}\ \mathcal{K}_{kn} - \mathcal{K}_{jk}\ \mathcal{K}_{mn}\big) \end{aligned} \tag{3.61}$$

Les termes soulignés correspondent à la valeur de Δ lorsque le tétraèdre est au repos ($L_{jk} = l_{jk}$). On montre que cette partie de Δ, divisée par $324\ \mathcal{V}^4$, est égale à $-\frac{3}{8}$. D'autre part, on peut vérifier que la somme des termes encadrés, une fois divisée par 324 $\mathcal{V}^4$, est égale à $-\frac{tr\,E}{2}$. Ces développements permettent de simplifier l'équation 3.60, pour obtenir :

$$\boxed{tr\ E^2 = \frac{1}{2}\big(tr\ E\big)^2 + \frac{\tilde{\Delta}}{324\ \mathcal{V}^4}} \tag{3.62}$$

où $\tilde{\Delta}$ prend la forme suivante :

$$\begin{aligned}
\tilde{\Delta} = \frac{1}{2} & \sum_{\text{arêtes}(j,k)} \big(L_{jk}^2 - l_{jk}^2\big)^2\ \mathcal{A}_j^2\ \mathcal{A}_k^2 \\
- & \sum_{\substack{\text{couples d'arêtes}\\((j,k),(j,m))}} \big(L_{jk}^2 - l_{jk}^2\big)\big(L_{jm}^2 - l_{jm}^2\big)\ \mathcal{A}_j^2\ \mathcal{K}_{km} \\
+ & \sum_{\substack{\text{couples d'arêtes}\\((j,k),(m,n))}} \big(L_{jk}^2 - l_{jk}^2\big)\big(L_{mn}^2 - l_{mn}^2\big)\ \big(\mathcal{K}_{jn}\ \mathcal{K}_{km} + \mathcal{K}_{jm}\ \mathcal{K}_{kn} - \mathcal{K}_{jk}\ \mathcal{K}_{mn}\big)
\end{aligned} \tag{3.63}$$

Finalement, l'énergie élastique de déformation s'exprime en fonction de la trace de E (équation 3.56) et de $\tilde{\Delta}$:

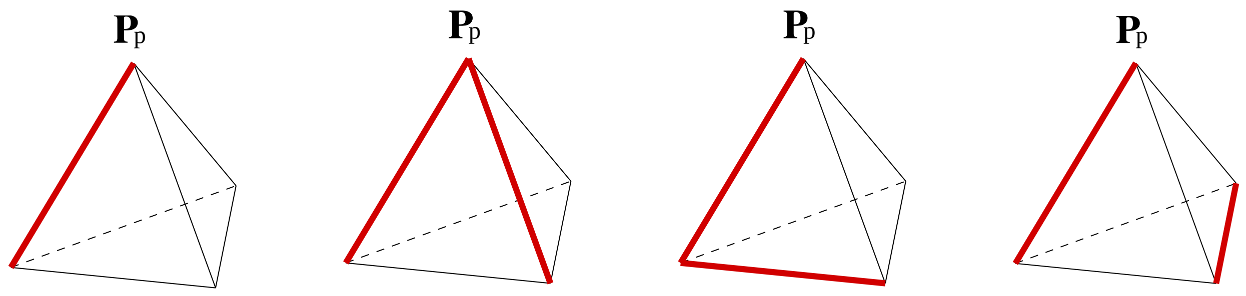
$$\boxed{W = \frac{1}{324\ \mathcal{V}^4}\left\{\left(\frac{\lambda+\mu}{2}\right)\left[\sum_{\text{Arêtes(j,k)}} \big(L_{jk}^2 - l_{jk}^2\big)\ \mathcal{K}_{jk}\right]^2 + \mu\ \tilde{\Delta}\right\}} \tag{3.64}$$

3.4.2.3 La force élastique

On obtient l'expression de la force élastique exercée par le tétraèdre sur un de ses sommets en dérivant l'énergie élastique par rapport à la position de ce sommet :

$$\begin{aligned}
\mathbf{F}_p = \frac{\partial W}{\partial \mathbf{Q}_p} = \frac{1}{162\ \mathcal{V}^4} \Bigg\{ & (\lambda+\mu) \left[\sum_{\text{Arêtes(j,k)}} \big(L_{jk}^2 - l_{jk}^2\big)\ \mathcal{K}_{jk}\right] \left(\sum_{\text{Arêtes(j,p)}} \overrightarrow{\mathbf{Q}_j\mathbf{Q}_p}\ \mathcal{K}_{jp}\right) \\
& + \mu \Bigg[\sum_{\text{Arêtes(j,p)}} \big(L_{jp}^2 - l_{jp}^2\big)\ \overrightarrow{\mathbf{Q}_j\mathbf{Q}_p}\ \mathcal{A}_j^2\ \mathcal{A}_p^2 \\
& \quad - \sum_{\substack{\text{couples d'arêtes}\\((j,p),(k,p))}} \Big[\big(L_{jp}^2 - l_{jp}^2\big)\ \overrightarrow{\mathbf{Q}_k\mathbf{Q}_p} + \big(L_{kp}^2 - l_{kp}^2\big)\ \overrightarrow{\mathbf{Q}_j\mathbf{Q}_p}\Big] \mathcal{A}_p^2\ \mathcal{K}_{jk} \\
& \quad - \sum_{\substack{\text{couples d'arêtes}\\((j,p),(j,k))}} \big(L_{jk}^2 - l_{jk}^2\big)\ \overrightarrow{\mathbf{Q}_j\mathbf{Q}_p}\ \mathcal{A}_j^2\ \mathcal{K}_{kp} \\
& \quad + \sum_{\substack{\text{couples d'arêtes}\\((j,k),(m,p))}} \big(L_{jk}^2 - l_{jk}^2\big)\ \overrightarrow{\mathbf{Q}_m\mathbf{Q}_p}\ \big(\mathcal{K}_{jp}\ \mathcal{K}_{km} + \mathcal{K}_{jm}\ \mathcal{K}_{kp} - \mathcal{K}_{jk}\ \mathcal{K}_{mp}\big) \Bigg] \Bigg\}
\end{aligned} \tag{3.65}$$

Cette formulation n'est pas très facile à utiliser, en particulier à cause des sommes sur les couples d'arêtes, qui doivent être interprétées comme indiqué sur la figure 3.18.



(a) Somme sur les arêtes (j,p) : 3 termes.

(b) Somme sur les couples d'arêtes ((j,p),(k,p)) : 3 termes.

(c) Somme sur les couples d'arêtes ((j,p),(j,k)) : 6 termes.

(d) Somme sur les couples d'arêtes ((j,k),(m,p)) : 3 termes.

FIG. 3.18 – *Sommes sur les arêtes et les couples d'arêtes d'un tétraèdre.*

En développant ces sommes et en mettant en facteur les variations de longueur de chaque arête du tétraèdre, on obtient une nouvelle expression présentée ci-dessous. Dans cette formulation, la somme sur les arêtes (j,p) signifie qu'il faut parcourir les 3 arêtes adjacentes au sommet $\mathbf{P}_p$ (sur lequel s'exerce la force), et la somme sur les arêtes (j,k), $j,k \neq p$ fait intervenir les 3 arêtes situées sur la face opposée au sommet $\mathbf{P}_p$.

$$\begin{aligned}\mathbf{F}_p = \frac{1}{162\,\mathcal{V}^4}\Bigg\{ &(\lambda+\mu)\left[\sum_{\text{Arêtes(j,k)}} \left(L_{jk}^2 - l_{jk}^2\right)\mathcal{K}_{jk}\right]\left(\sum_{\text{Arêtes(j,p)}} \overrightarrow{\mathbf{Q}_j\mathbf{Q}_p}\,\mathcal{K}_{jp}\right) \\ &+\mu\Bigg[\sum_{\text{Arêtes(j,p)}} \left(L_{jp}^2 - l_{jp}^2\right)\mathcal{A}_p^2\left(\overrightarrow{\mathbf{Q}_j\mathbf{Q}_p}\,\mathcal{A}_j^2 - \overrightarrow{\mathbf{Q}_k\mathbf{Q}_p}\,\mathcal{K}_{jk} - \overrightarrow{\mathbf{Q}_m\mathbf{Q}_p}\,\mathcal{K}_{jm}\right) \\ &+\sum_{\substack{\text{Arêtes(j,k)}\\ j,k\neq p}} \left(L_{jk}^2 - l_{jk}^2\right)\left(\overrightarrow{\mathbf{Q}_m\mathbf{Q}_p}\,\mathcal{K}_{jkmp} - \overrightarrow{\mathbf{Q}_j\mathbf{Q}_p}\,\mathcal{A}_j^2\mathcal{K}_{kp} - \overrightarrow{\mathbf{Q}_k\mathbf{Q}_p}\,\mathcal{A}_k^2\mathcal{K}_{jp}\right)\Bigg]\Bigg\}\end{aligned} \tag{3.66}$$

avec :

- dans la deuxième ligne, k et m sont les indices complémentaires de j et p,
- dans la dernière ligne, m est l'indice complémentaire de j, k et p, et

$$\mathcal{K}_{jkmp} = \mathcal{K}_{jp}\,\mathcal{K}_{km} + \mathcal{K}_{jm}\,\mathcal{K}_{kp} - \mathcal{K}_{jk}\,\mathcal{K}_{mp}$$

On obtient, au final, une formulation de la force élastique exercée sur chaque sommet du tétraèdre, qui ne dépend que de la variation de longueur des arêtes du tétraèdre, de scalaires qui représentent la forme au repos du tétraèdre, et des vecteurs d'arêtes. Cette expression est donc très proche de ce qui est obtenu avec un modèle masse-ressort. Dans notre mise en œuvre actuelle, les performances de ce nouveau modèle sont équivalentes à celles du masse-tenseur non-linéaire présenté dans la première partie de ce chapitre. Cependant, nous pensons améliorer ses performances en optimisant la structure de données, en particulier au niveau de la façon dont nous effectuons les boucles sur les couples d'arêtes d'un tétraèdre.

3.5 Conclusion

Nous avons consacré ce chapitre à l'étude de modèles déformables basés sur l'élasticité de St Venant-Kirchhoff, qui est la généralisation en grands déplacements de l'élasticité linéaire. Ces modèles déformables ont la particularité d'être invariants par rotation, ce qui améliore nettement le réalisme des déformations dès que le seuil des petits déplacements est franchi.

Nous avons proposé deux formulations différentes du même modèle. La première est une extension du modèle masse-tenseur linéaire du chapitre 2, que nous avons ensuite complété avec des contraintes d'incompressibilité. Nous avons montré l'intérêt que présente ce modèle non-linéaire en le comparant au modèle linéaire dans le cas de déformations faisant intervenir de grands déplacements. Mais le réalisme supérieur de ce nouveau modèle se paye en terme de performance, ce qui nous a conduit à introduire un modèle adaptatif, dont les zones sont traitées de manière linéaire ou non-linéaire suivant que le déplacement qu'elles subissent est considéré comme petit ou grand. Finalement, afin de transférer toutes nos améliorations du linéaire vers le non linéaire, nous avons étendu le comportement transversalement isotrope à l'élasticité de St Venant Kirchhoff.

Dans la seconde moitié du chapitre, nous avons introduit une nouvelle formulation du même modèle, en partant de l'idée qu'il devait être possible d'intégrer la propriété d'invariance par transformation rigide dans la formulation de l'énergie élastique. Nous obtenons ainsi un modèle déformable dont l'énergie et les forces élastiques dépendent uniquement des variations de longueur des arêtes des tétraèdres (ou des triangles dans le cas 2D) qui composent le maillage, de leur formes au repos et des cœfficients d'élasticité du matériau.

Chapitre 4

Interactions avec les modèles déformables

Dans les deux chapitres précédents, nous avons développé des modèles déformables qui nous permettent de simuler la déformation et la découpe d'organes du corps humain. Mais cela ne suffit pas pour simuler une opération chirurgicale. Il faut aussi être capable d'interagir avec ces modèles déformables à l'aide d'instruments virtuels. Nous allons donc expliquer, dans ce chapitre, comment nous modélisons les contacts entre des instruments chirurgicaux et les organes virtuels. Nous verrons ensuite par quels moyens nous avons couplé notre simulateur avec un dispositif à retour d'effort.

4.1 Introduction

Un simulateur de chirurgie est composé de deux parties principales. La première partie est la boucle de simulation de l'organe virtuel. Cette boucle comprend l'application de contraintes sur le modèle déformable et le calcul des déformations qui en résultent. Nous avons présenté, dans les trois premiers chapitres de cette thèse, les modèles déformables qui vont nous permettre de simuler, en temps réel, les déformations et la découpe d'organes mous. La seconde partie du simulateur est l'interface grâce à laquelle l'utilisateur va pouvoir d'interagir avec l'organe virtuel. Dans le cadre de la simulation de chirurgie laparoscopique sur le foie, cette interaction se fait par l'intermédiaire de longs outils qui ont la particularité de passer par un point fixe correspondant à l'endroit où l'instrument passe au travers de l'abdomen du patient (voir paragraphe 2 de l'introduction). Nous utilisons pour cela des "Laparoscopic Impulse Engines" (LIE), développés par la société américaine *Immersion Corporation*[1], qui ont l'avantage d'avoir une ergonomie très proche de celle des véritables outils utilisés par les chirurgiens. De plus, ces instruments sont équipés d'un système à retour d'effort permettant de donner à l'utilisateur la sensation de toucher l'organe virtuel.

Le développement de l'interface permettant à un utilisateur d'effectuer certains gestes chirurgicaux sur l'organe virtuel peut se décomposer en deux étapes. La première consiste à modéliser les interactions entre l'outil et le modèle déformable. On commence alors par déterminer si l'outil manipulé par l'utilisateur est en contact avec l'organe virtuel. Si c'est le cas, il faut alors modéliser précisément le contact et les interactions éventuelles (glisser, agripper, découper, ...). De plus, il faut calculer la force exercée par le modèle déformable sur l'outil.

La seconde partie est le couplage avec un système à retour d'effort, qui permettra à l'utilisateur de diriger l'instrument de chirurgie virtuel et de "toucher" le modèle déformable. Pour cela, il faut définir un protocole de communication entre la boucle de simulation et le système à retour d'effort. En effet, le modèle déformable et le système à retour d'effort ont des fréquences de fonctionnement différentes, ce qui nécessite l'utilisation de deux boucles asynchrones. Comme c'est le cas dans la plupart des simulateurs utilisant un système à retour d'effort, nous avons séparé ces deux boucles sur deux ordinateurs différents. Une communication bidirectionnelle va alors permettre d'envoyer la position de l'outil à la boucle de simulation qui enverra en échange la force qui doit être appliquée sur l'instrument. La gestion de ce retour d'effort pose, en général, quelques problèmes, principalement dus aux fréquences de simulation mises en jeu et à la synchronisation entre les processus de rendu visuels et haptiques. On peut trouver un grand nombre de travaux sur ce sujet dans les actes des *"Phantom Users Group Workshop"*, une série de conférences regroupant chaque année les utilisateurs du plus utilisé des systèmes à retour d'effort, le *Phantom*, dont nous parlerons plus en détail au paragraphe 4.3 (Salisbury et Srinivasan, 1996; Salisbury et Srinivasan, 1997a; Salisbury et Srinivasan, 1998; Salisbury et Srinivasan, 1999; Salisbury et Srinivasan, 2000). Le site internet de l'*Haptics Community*[2] regroupe aussi un grand nombres de publications et tient à jour des bases

1. http://www.immerse.com/lapimpulseengine.html
2. http://haptic.mech.northwestern.edu/

de données de laboratoires et de chercheurs impliqués dans les différentes domaines d'applications du retour d'effort. C'est une véritable mine d'informations.

Dans ce chapitre, nous présenterons dans un premier temps la façon dont nous détectons et traitons les contacts entre le modèle déformable et les instruments chirurgicaux. Nous détaillerons ensuite la modélisation de quelques gestes et nous nous attarderons un peu sur le problème de la découpe. Nous décrirons ensuite le protocole de communication que nous avons établi entre l'ordinateur utilisé pour le modèle déformable et celui qui gère le système à retour d'effort. Enfin, nous proposerons notre solution au problème du retour d'effort.

4.2 Détection et gestion des collisions

Dans un simulateur de chirurgie, les déformations subies par l'organe virtuel sont le plus souvent engendrées par le contact avec les instruments chirurgicaux. Le réalisme de la simulation dépend donc directement de la qualité de la modélisation des contacts. Nous allons décomposer cette modélisation en trois parties. Tout d'abord, il faut détecter une éventuelle collision entre la surface de l'organe virtuel et les outils chirurgicaux. En cas de collision, on doit connaître la zone concernée de la surface, c'est-à-dire un ensemble de sommets, d'arêtes et de faces. La seconde étape consiste alors à appliquer des contraintes sur cette partie de la surface de manière à déformer l'organe virtuel. De ces contraintes dépendra l'aspect du contact. Nous verrons enfin comment enrichir le traitement des collisions afin de modéliser quelques gestes de base qui sont utilisés couramment par les chirurgiens.

La figure 4.1 montre la représentation des instruments de chirurgie laparoscopique au sein de notre simulateur. Ils consistent en un long et fin cylindre, terminé par une pince ou un crochet. Ces instruments ont la particularité de passer par un point fixe qui correspond à son point introduction au travers de l'abdomen du patient.

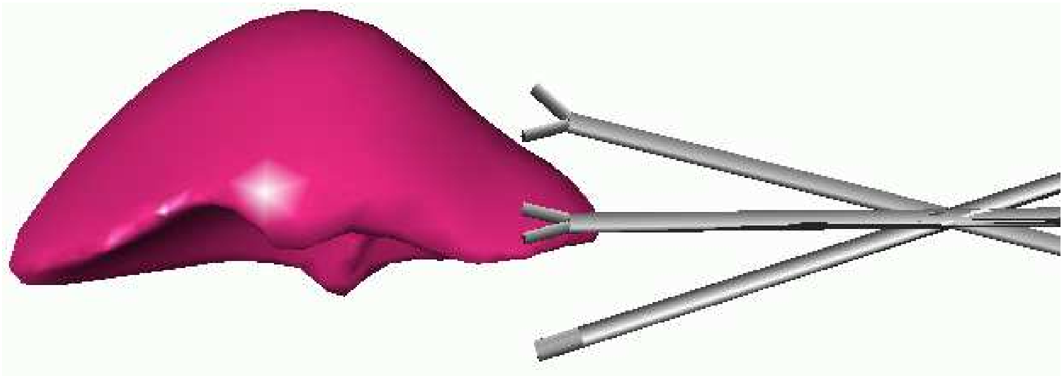


FIG. 4.1 – *Modélisation des instruments de chirurgie laparoscopique.*

4.2.1 Détection de collision : la méthode LCN

La détection de collision est un problème bien connu en animation par ordinateur. Lin et Gottschalk proposent dans (Lin et Gottschalk, 1998), un récapitulatif assez complet des différentes méthodes développées ces dernières années, et les classent en fonction du type de représentation des objets (polyèdres, surfaces implicites ou paramétriques, etc...). Les modèles déformables que nous avons développés possèdent tous une surface triangulée, qui sert notamment pour leur affichage à l'écran. D'autre part, il est possible de représenter les instruments de chirurgie que nous utilisons par quelques polygones. Nous nous intéressons donc aux méthodes permettant de détecter l'interpénétration de deux modèles polygonaux. Des méthodes très efficaces ont été développées pour la détection de collision entre des polyèdres convexes (Lin et Canny, 1992; Baraff, 1990), mais elles s'appliquent malheureusement à un nombre restreint d'objets. Pour les polyèdres de formes plus générales, les optimisations reposent le plus souvent sur le pré-calcul d'une hiérarchie de volumes englobants (Gottschalk et al., 1996), mais ces méthodes perdent leurs avantages lorsque les objets bougent ou se déforment. De plus, nous avons besoin d'une détection de collision qui soit dynamique, dans la mesure où les instruments peuvent avoir des mouvements relativement rapides. Il faut donc pouvoir considérer le volume 4D défini par le déplacement de l'instrument entre deux instants. C'est pourquoi nous utilisons la méthode LCN (Lombardo et al., 1999).

Cette méthode a été développée durant l'action incitative AISIM[3], par J.-C. Lombardo (projet Epidaure, INRIA Sophia Antipolis), M.-P. Cani et F. Neyret (projet iMAGIS, INRIA Rhône-Alpes). L'idée principale est d'utiliser la librairie graphique OpenGL pour détecter l'intersection entre une scène complexe (le ou les modèles déformables) et un objet polyédrique composé d'un nombre limité de faces (l'outil de laparoscopie). L'objet polyédrique est alors considéré comme une caméra OpenGL à laquelle on ajoute un ensemble de plans de coupes (c'est le nombre de plans de coupes disponibles pour une caméra qui limite la complexité de l'objet). La collision est détectée simplement en appelant la première partie du processus de rendu de la caméra, qui consiste à sélectionner les primitives géométriques qui sont dans son champ de vue. On obtient ainsi les triangles de la surface qui sont intersectés par l'outil. On peut, de cette manière, détecter la collision avec un outil considéré comme statique en utilisant une caméra orthographique, dont le champ de vue est un parallélépipède. Mais on peut aussi utiliser une caméra en géométrie perspective dont le champ de vue de forme pyramidale permettra, à l'aide de quelques plans de coupe, de représenter le volume décrit par l'outil entre deux pas de temps (pour plus de détails se reporter à l'article (Lombardo et al., 1999)).

Cette méthode est très bien adaptée à notre problème puisque l'instrument de laparoscopie peut être approximé par un simple parallélépipède. De plus, elle présente deux avantages. Premièrement, elle est relativement simple à programmer, puisque environ une vingtaine de lignes de code OpenGL suffit. Deuxièmement, cette méthode est très efficace, en particulier lorsque l'on utilise des cartes graphiques accélératrices. Par exemple, sur une SGI Onyx2 IR, la collision entre l'outil et le modèle de foie (dont la surface possède 1224 faces) prend moins de 0.2 ms. Même sur un PC Pentium II

3. Action Incitative SIMulation de chirurgie : http://www-sop.inria.fr/epidaure/AISIM/

333MHz sous linux, sans carte accélératrice 3D, et en utilisant la librairie *software* Mesa 3D[4], la détection de collision dure moins de 3.0 ms.

La figure 4.2 montre le résultat de la détection de collision entre l'outil chirurgical en mouvement (il est représenté à deux instants successifs) et la surface du maillage représentant le foie. L'algorithme retourne l'ensemble des faces intersectées.

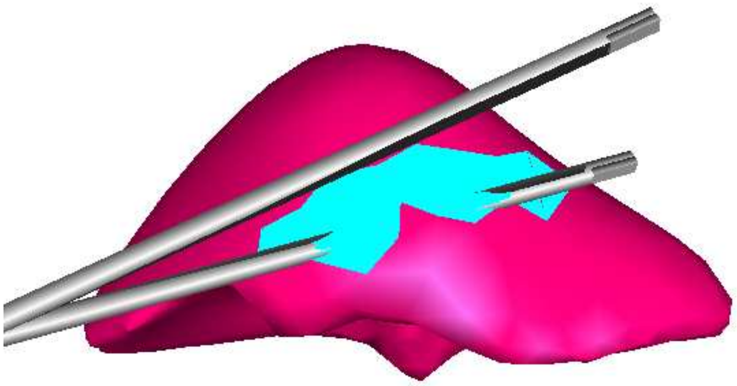


FIG. 4.2 – *Détection de collision entre l'outil chirurgical en mouvement et un modèle de foie.*

4.2.2 Modélisation des contacts

Une fois que l'on connaît la liste des triangles de la surface du foie qui sont en collision avec l'instrument de chirurgie, il faut appliquer à ces faces des contraintes afin de les repousser en dehors de l'instrument et de modéliser le contact. Une technique souvent utilisée est la méthode des pénalités. Elle consiste à appliquer sur la zone en collision des contraintes sous la forme de forces proportionnelles à la profondeur d'interpénétration. Cette partie du modèle déformable va alors se déformer pour "ressortir" de l'objet avec lequel elle est en collision.

Le problème de cette méthode est qu'elle ne respecte pas vraiment la géométrie du contact. En effet, rien ne garantit que les faces en collision vont suivre exactement les déplacements de l'outil. Suivant le réglage des forces de pénalité, certaines faces peuvent rester à l'intérieur de l'instrument durant plusieurs pas de temps, ou bien au contraire risquent d'être repoussées trop fort, ce qui ne manquera pas de créer des discontinuités et des problèmes de stabilité. En fait, cette méthode ne fonctionnera bien que si la fréquence de simulation est suffisamment élevée. Or nous avons pris le parti de développer des modèles déformables dont le comportement biomécanique est relativement élaboré, et que nous ne pouvons pas nous permettre de faire fonctionner à des fréquences trop importantes.

Nous avons donc choisi une approche plus directe, fondée sur des contraintes géométriques : nous appliquons des déplacements précis à l'ensemble des faces intersectées. L'idée principale de

4. http://www.mesa3d.org/

notre approche est de calculer ces déplacements de telle manière que tous les triangles restent, à chaque pas de temps, parfaitement en contact avec l'instrument. Ceci permet d'éviter que le modèle déformable ne se mette à osciller entre l'intérieur et l'extérieur de l'instrument. Enfin, nous gardons à l'esprit que les calculs de ces déplacements doivent être les plus simples possibles afin de ne pas pénaliser la fréquence globale de simulation.

Dans de nombreux travaux, le contact entre un instrument rigide et un objet mou est souvent limité à un point situé au bout de l'instrument ou à un nombre restreint de points d'échantillonnage répartis le long de l'instrument. La raison principale de telles simplifications étant que le temps de calcul de l'algorithme de détection de collision est directement proportionnel au nombre de points considérés. À l'inverse, la méthode de détection de collision que nous utilisons est très efficace, et surtout permet de tenir compte du volume total de l'outil. Nous allons donc mettre à profit ces avantages pour modéliser le plus précisément possible le contact entre l'outil tout entier et la surface de l'organe.

Nous voulons en particulier que le contact entre le manche de l'outil et la surface de l'organe soit pris en compte de manière très réaliste. Le problème se ramène alors à déplacer un ensemble de triangles adjacents afin qu'ils viennent tous s'appuyer sur la même droite qui représente le bord de l'instrument. Ces triangles ayant tous des arêtes en commun deux à deux, il est facile de montrer qu'ils appartiendront alors à un même plan. La solution est donc de les projeter sur un plan s'appuyant sur le manche de l'outil. Pour cela, on procède de la manière suivante :

- On considère une droite ($\mathbf{F}$,$\mathbf{B}$) représentant le bord du manche de l'instrument ($\mathbf{F}$ correspond au point fixe de l'outil et $\mathbf{B}$ à son extrémité), et l'ensemble des faces en collision avec le manche de l'instrument (figure 4.3 en haut à gauche).
- À partir de la moyenne $\vec{\mathbf{N}}$ des normales aux faces intersectées et de $\vec{\mathbf{T}}_{\mathbf{FB}}$, le vecteur directeur de la droite ($\mathbf{F}$,$\mathbf{B}$), on calcule la normale au plan de projection par :

$$\vec{\mathbf{N}}_{Proj} = \left(\vec{\mathbf{T}}_{\mathbf{FB}} \wedge \vec{\mathbf{N}}\right) \wedge \vec{\mathbf{T}}_{\mathbf{FB}}. \tag{4.1}$$

- Le plan de projection est alors défini par la droite ($\mathbf{F}$,$\mathbf{B}$) et le vecteur normal $\vec{\mathbf{N}}_{Proj}$. Le déplacement de chaque sommet intersecté correspond alors à sa projection orthogonale sur ce plan (figure 4.3 en bas et en haut à droite, figure 4.5(a)) :

$$d\vec{\mathbf{P}} = \left(\vec{\mathbf{PB}} \,.\, \vec{\mathbf{N}}_{Proj}\right) \vec{\mathbf{N}}_{Proj}. \tag{4.2}$$

Mais il y a deux problèmes avec cette méthode. D'abord, elle ne s'applique pas bien quand la direction de l'instrument n'est plus tangente à la surface du foie. En effet, dans ce cas, seule l'extrémité de l'instrument est en collision avec la surface. L'application de la méthode décrite ci-dessus provoque alors une déformation peu réaliste (voir figure 4.4). Le second problème vient du fait que, quand l'outil est perpendiculaire à la surface, le plan de projection est mal défini (produit vectoriel de deux vecteurs parallèles). En fait, ces deux problèmes n'en sont qu'un : la méthode ne fonctionne plus dès qu'il s'agit de modéliser le contact avec l'extrémité de l'instrument. En effet,

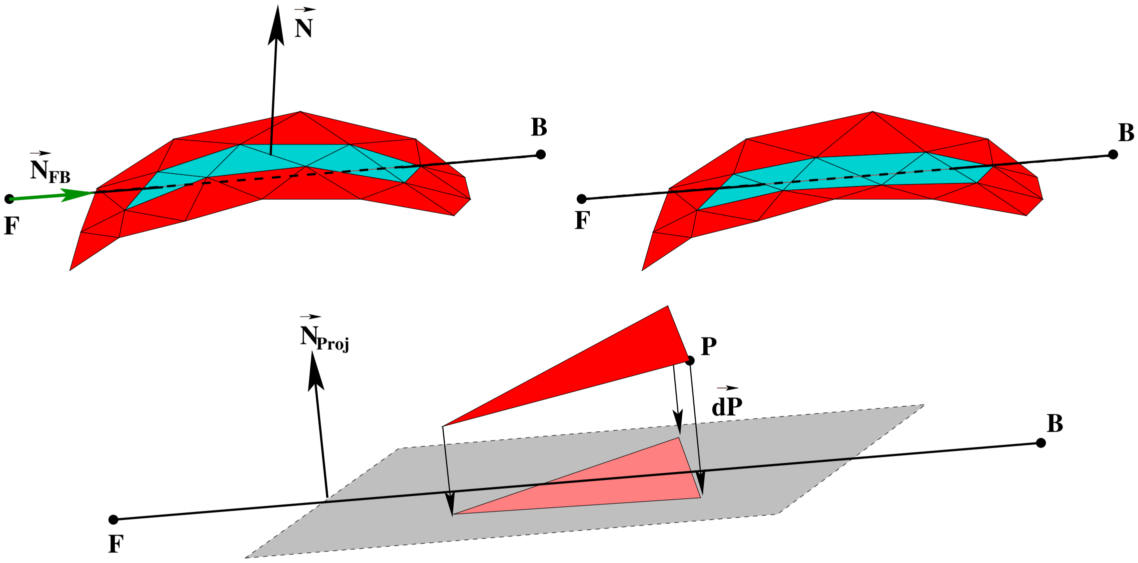


Fig. 4.3 – *Définition du plan de projection, et projection des faces intersectées sur celui-ci.*

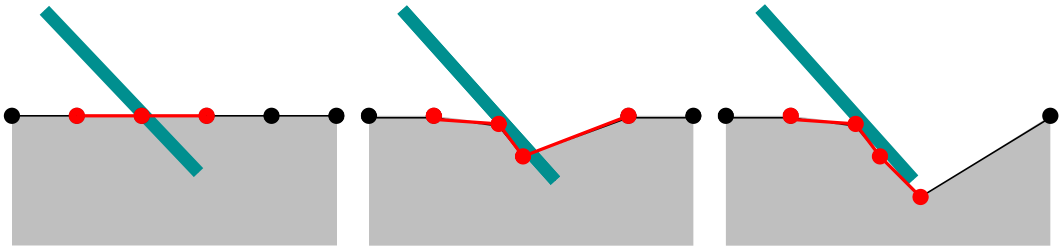
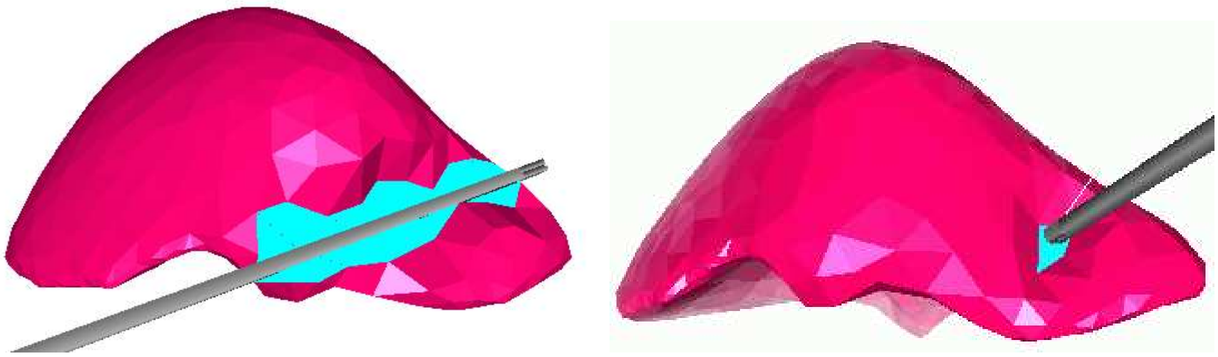


Fig. 4.4 – *Problème rencontré lorsque le extrémité de l'instrument est en contact avec la surface.*



(a) Contact avec le manche de l'instrument.

(b) Contact avec le bout de l'instrument.

Fig. 4.5 – *Modélisation du contact entre un instrument chirurgical et un modèle déformable représentant le foie.*

les faces intersectées par l'extrémité de l'instrument doivent simplement être projetées sur le plan défini par le point **B** et la moyenne des normales aux faces intersectées (figure 4.5(b)).

Malheureusement, il arrive souvent que le manche et l'extrémité de l'instrument soient simultanément en contact avec la surface de l'organe. Il est alors difficile de séparer l'ensemble des faces intersectées en deux, un pour le bout et un pour le manche. La solution est de mettre une fois de plus à profit l'efficacité de l'algorithme de détection de collision : au lieu de faire une détection de collision avec l'ensemble de l'outil, nous séparons le manche de l'instrument de son extrémité et nous faisons une détection de collision pour chaque partie de l'outil. La première détection de collision donne l'ensemble des faces qui sont en collision avec le bout de l'instrument (environ 1 cm), et la deuxième fournit les faces en contact avec le manche (et qui ne font pas partie du premier ensemble). On peut alors appliquer un traitement spécifique à chaque ensemble de triangles (figures 4.6 et 4.7).

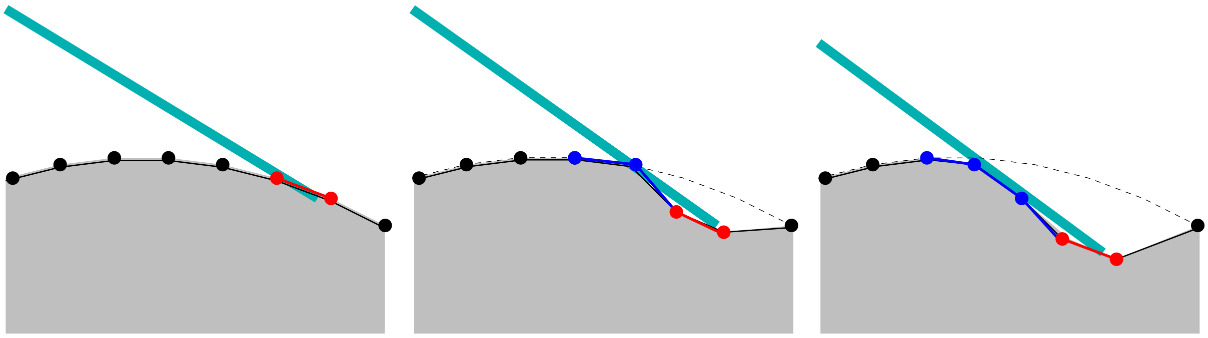


FIG. 4.6 – *La double détection de collision permet de traiter différemment les faces en collision avec l'extrémité de l'instrument (en rouge) et celles en contact avec le manche (en bleu).*

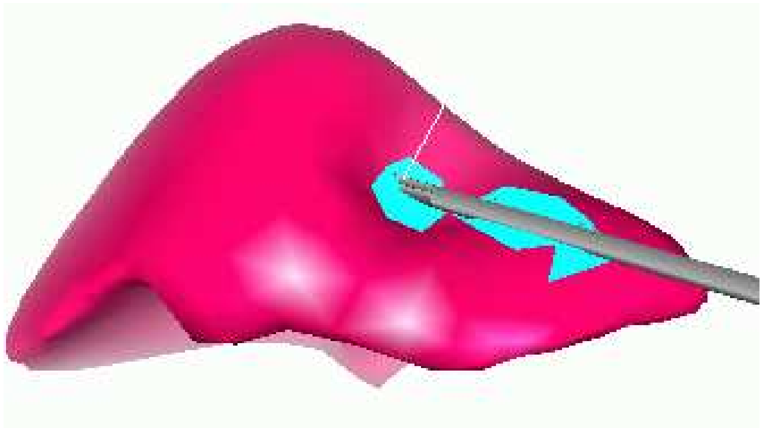
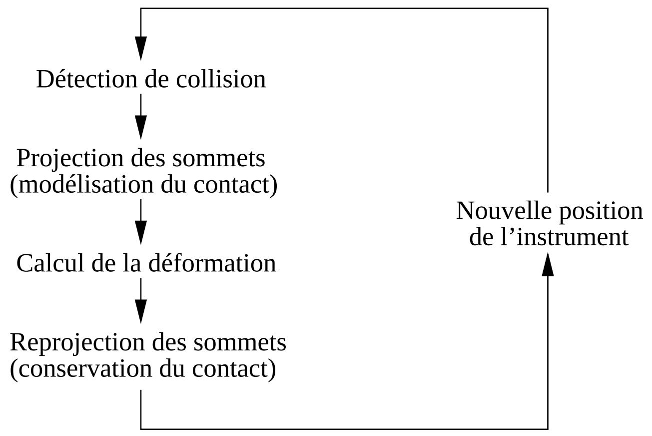


FIG. 4.7 – *Modélisation d'un contact entre le foie et l'instrument du chirurgien obtenue en traitant séparément les contacts avec le manche et avec l'extrémité.*

La capsule de Glisson qui constitue la surface du foie est particulièrement glissante. Nous voulons que le comportement de base de l'instrument de chirurgie sur cette surface soit le glissement parfait. C'est pour cette raison que les projections que nous venons de décrire se font selon une direction orthogonale à la surface, indépendante du déplacement et de la vitesse de l'instrument. Pour cette même raison, nous ne voulons pas appliquer de contraintes de position sur les sommets que nous projetons. En effet, après l'application des déplacements permettant de modéliser le contact avec l'instrument, les sommets de la surface du foie demeurent libres. Ils vont donc se déplacer lors du calcul de la déformation, mais le contact sera conservé en les reprojetant sur leurs plans respectifs. On obtient alors le schéma de simulation suivant :



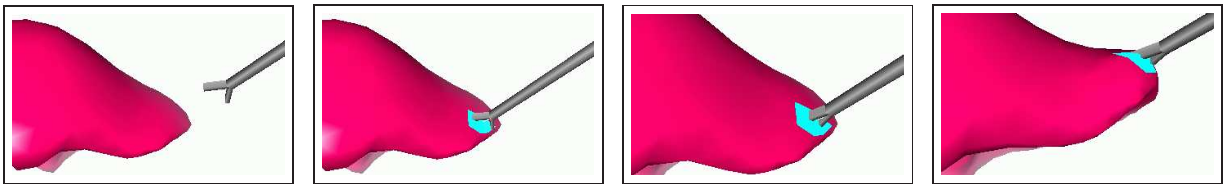
Cette technique permet non seulement à l'instrument de glisser sur la surface, mais aussi aux sommets de la surface de glisser sous l'instrument si les contraintes mécaniques subis par le matériau les y obligent.

4.2.3 Modélisation de quelques gestes élémentaires

Les interactions entre un outil de chirurgie laparoscopique et un organe comme le foie sont dans la réalité très complexes, et donc relativement difficiles à simuler de manière réaliste. Pourtant, dans un simulateur de chirurgie, la qualité de ces interactions va fortement influencer le réalisme global de la simulation. C'est pourquoi nous nous sommes attachés à modéliser quelques-uns des gestes les plus souvent utilisés dans ce type de chirurgie :

- **Glisser sur la surface :** c'est le comportement de base de nos outils. Les chirurgiens utilisent cette propriété afin de glisser des instruments entre les différents organes, ou encore, dans le cas du foie, pour en soulever une partie afin d'accéder à une zone cachée de l'anatomie du patient. Dans notre simulateur, le manche de l'outil aura toujours cette propriété de glisser sur la surface des organes. Par contre, nous avons prévu d'autres possibilités pour l'extrémité de l'instrument.
- **Agripper une partie du foie :** Un des outils souvent utilisés est terminé par une pince munie de dents. Même si la surface du foie est particulièrement glissante, quelques parties

peuvent tout de même être agrippées par un tel instrument. Durant une opération, il arrive souvent que le chirurgien saisisse le bord du foie avec cette pince afin de le soulever ou pour écarter un morceau qu'il est en train de découper. Nous avons modélisé ce geste de la manière suivante (voir figure 4.8) : lorsque l'extrémité de la pince entre en contact avec le foie, avec ses mâchoires grandes ouvertes, et que l'utilisateur les ferme alors à moitié, nous considérons que les triangles en collision avec le bout de l'instrument y sont accrochés. Ces triangles subiront alors les mêmes déplacements que l'extrémité de l'instrument jusqu'à ce que les mâchoires de la pince soient reouvertes afin de lâcher prise.



(a) La pince grande ouverte...

(b) ...entre en collision avec le foie...

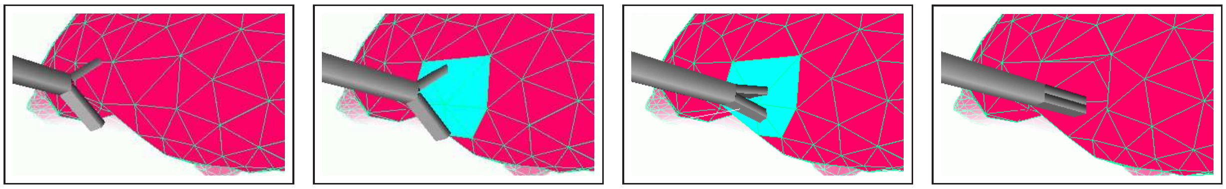
(c) ...et se referme à moitié.

(d) La partie du foie en contact avec l'extrémité de la pince est alors agrippée.

FIG. 4.8 – *Modélisation de la mâchoire d'une pince saisissant le bord du foie.*

- **Découper le foie par retrait de matière :** un des avantages importants de nos modèles déformables masse-tenseurs (linéaires, anisotropes ou en grands déplacements), est de se prêter parfaitement à la simulation de découpe. Lors d'opérations chirurgicales sur le foie, le chirurgien n'effectue pas les incisions avec un scalpel. En effet, le foie étant fortement vascularisé, il risquerait alors de sectionner des vaisseaux importants et de provoquer une hémorragie. La découpe du parenchyme hépatique se fait plutôt par "retrait de matière". Nous avons visualisé des enregistrements d'interventions chirurgicales sur le foie, durant lesquelles les chirurgiens utilisaient deux techniques principales pour effectuer des découpes. La première consiste simplement à écraser le parenchyme hépatique entre les mâchoires d'une pince. Le parenchyme étant relativement friable, cela permet de voir apparaître les vaisseaux sans risquer de les endommager. Nous simulons ce geste de manière presque identique au cas où les mâchoires de la pince agrippent le foie. Si au lieu de fermer à moitié la mâchoire de la pince, l'utilisateur la ferme totalement, les tétraèdres correspondant aux faces en collision avec l'extrémité de l'instrument sont supprimés (voir figure 4.9).

  La seconde méthode utilisée par les chirurgiens pour découper le parenchyme hépatique met en œuvre un autre instrument : le cavitron, ou bistouri à ultrasons. L'extrémité de l'instrument émet des ultrasons qui font exploser les cellules du foie. Le principal avantage de cet instrument, outre sa simplicité d'utilisation, est que si la sonde à ultrasons est réglée à la bonne fréquence, ils n'endommageront pas les parois des vaisseaux sanguins. Nous simulons l'effet de



(a) La pince grande ouverte...

(b) ...entre en collision avec le foie...

(c) ...et se referme complétement.

(d) Les tétraèdres en contact avec la pince sont alors supprimés.

FIG. 4.9 – *Simulation de découpe par retrait des tétraèdres écrasés par les mâchoires de la pince.*

cet instrument simplement en supprimant tous les tétraèdres en contact avec son extrémité.

La séquence d'images de la figure 4.10, montre l'utilisation des gestes que nous venons de décrire. Deux instruments sont utilisés. Le premier vient agripper la partie droite du foie et exerce une force vers la droite, tandis que le second, qui simule un bistouri à ultrasons vient découper le parenchyme hépatique.

4.2.4 Amélioration de la découpe

La méthode de découpe par "retrait de matière" que nous venons de définir donne de très bons résultats si les tétraèdres de la zone concernée sont suffisamment petits. C'est par exemple le cas sur la figure 4.10, pour laquelle nous avons au préalable raffiné la partie de foie dans laquelle nous découpons. On obtient alors un aspect granuleux de la surface découpée qui est relativement réaliste. Par contre, si les tétraèdres sont trop gros, on voit apparaître, au fur et à mesure de la découpe, les arêtes vives des tétraèdres situés à l'intérieur du foie. Comme il n'est pas envisageable de travailler avec des maillages trop fins qui rendraient les calculs trop lents, il nous a fallu améliorer notre méthode de découpe.

Des travaux intéressants sont menés sur la découpe de maillage. Le principe en est toujours le même : la trajectoire d'un outil définit un plan (ou une surface) qui intersecte une partie du maillage. Il s'agit alors de remailler la zone en découpant toutes les primitives géométriques traversées par le plan de coupe. Dans le cas de la découpe d'un modèle déformable, il faut ensuite remettre à jour les propriétés locales du matériau. Si on utilise un modèle masse-ressort, il faut calculer les raideurs des nouveaux ressorts et mettre à jour la répartition des masses (Ganovelli et al., 2000b; Ganovelli et al., 2000a; Bielser et al., 1999). Pour un modèle éléments finis (Mor et Kanade, 2000), il faut recalculer les matrices de rigidité locales comme nous l'avons expliqué au paragraphe 2.4.2.4. La principale difficulté de ces techniques est de créer un minimum de nouveaux éléments afin de ne pas trop complexifier le modèle, mais aussi de suivre, de manière continue, l'évolution de la découpe (Mor et Kanade, 2000).

Mais ces méthodes permettent de découper un maillage à la manière d'un scalpel et non pas en supprimant une partie du volume. Pour cela, il faudrait donner une épaisseur plan de coupe, ce

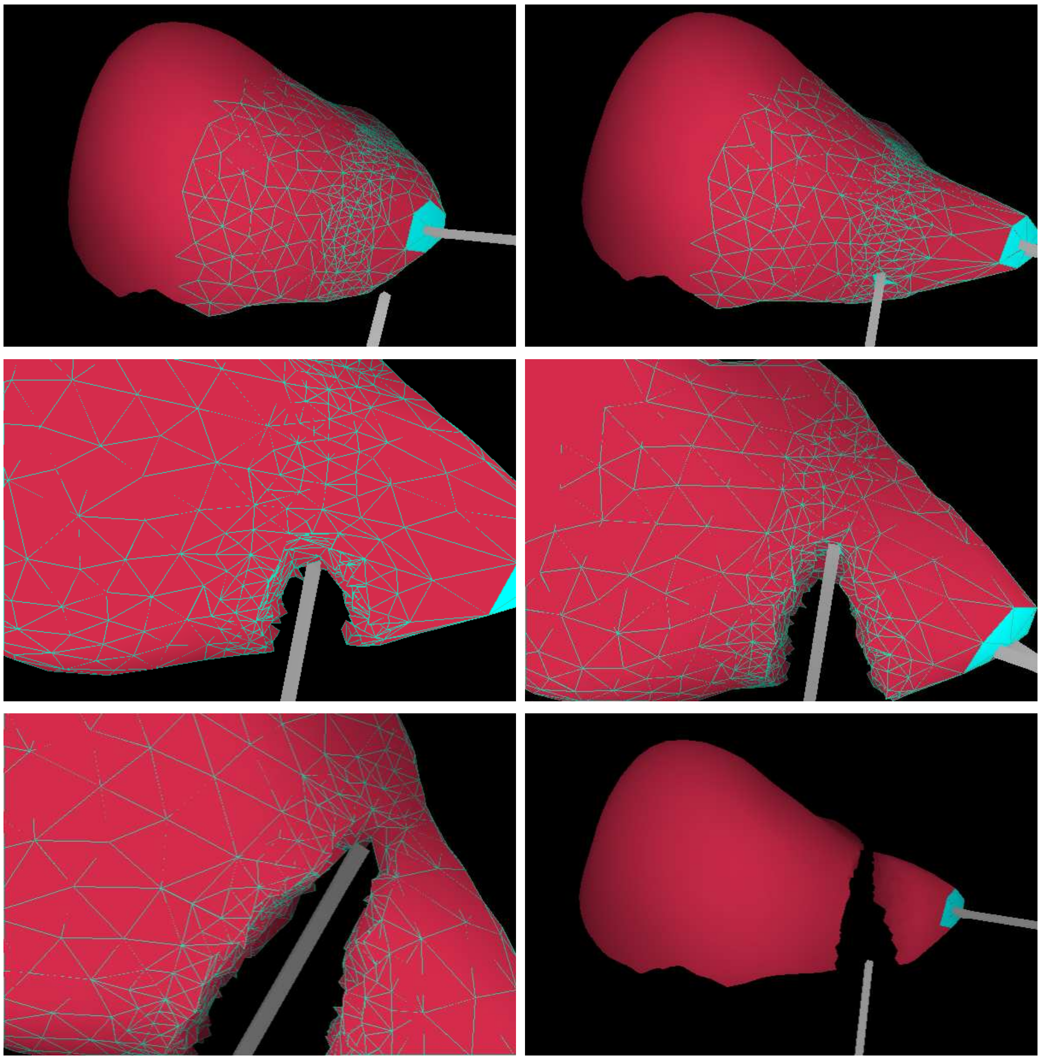


FIG. 4.10 – *Simulation d'une hépatectomie.*

qui compliquerait encore le problème. Il serait plus simple, dans notre cas, de raffiner localement le maillage afin d'obtenir des tétraèdres plus petits qui donneraient un meilleur aspect à la découpe. On se rapproche alors du problème du maillage adaptatif. Le principe consiste à définir un maillage avec plusieurs résolutions et de pouvoir choisir à tout moment et sur n'importe quelle zone, la résolution la plus adaptée. Par exemple pour calculer la déformation d'un objet mou, il est intéressant de pouvoir choisir un maillage fin dans la zone où on applique des contraintes, afin d'augmenter la précision, et un maillage plus "grossier" dans les zones où il ne se passe presque rien afin de gagner du temps. Ce type de méthodes est particulièrement bien adapté aux maillages cubiques réguliers (Ganovelli et al., 1999; Debunne et al., 2000) (ou carrés en dimension 2 (Hutchinson et al., 1996)), mais peut aussi être utiliser sur des maillages tétraédriques (Astley et Hayward, 1997; Debunne et al., 2000). Par contre, les structures de données complexes qui sont nécessaires à la gestion des différents niveaux de résolution rendent les modifications topologiques délicates et coûteuses.

Notre problème est un peu plus simple. On voudrait effectivement être capable de rendre le maillage plus fin dans la zone où l'on est en train de découper. Par contre, par rapport à un maillage adaptatif, nous n'avons pas besoin de garder une trace de l'ancienne configuration du maillage. Nous ne ferons en effet jamais le cheminement inverse pour revenir à un maillage plus grossier. Nous avons donc juste besoin d'une opération topologique peu coûteuse permettant de diminuer localement la taille des tétraèdres du maillage. Nous avons choisi une opération qui consiste à couper une arête en deux. Cela revient à diviser en deux tous les tétraèdres qui contiennent cette arête (figure 4.11). Nous avons aussi mené des travaux pour optimiser la position du nouveau sommet sur l'arête en fonction de la qualité[5] des tétraèdres qui vont être créés (Péquignot, 1999).

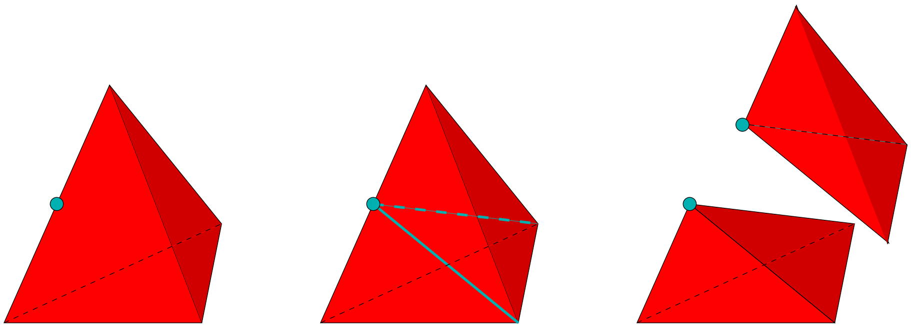


FIG. 4.11 – *Division d'un tétraèdre en deux en introduisant un nouveau point sur une de ses arêtes.*

La figure 4.12 montre le résultat obtenu en subdivisant successivement les arêtes sur une zone de la surface du maillage.

Notre but étant de raffiner le maillage dans la zone où on va le découper, nous devons coupler le mécanisme de division d'arête avec l'outil manipulé par l'utilisateur. Pour cela, nous ajoutons une étape juste avant la détection et la gestion des collisions. Cette étape consiste à effectuer une première détection de collision entre l'outil et la surface du maillage afin de déterminer les faces

5. On entend par "qualité" d'un tétraèdre la régularité de sa forme, le tétraèdre parfait étant équilatéral.

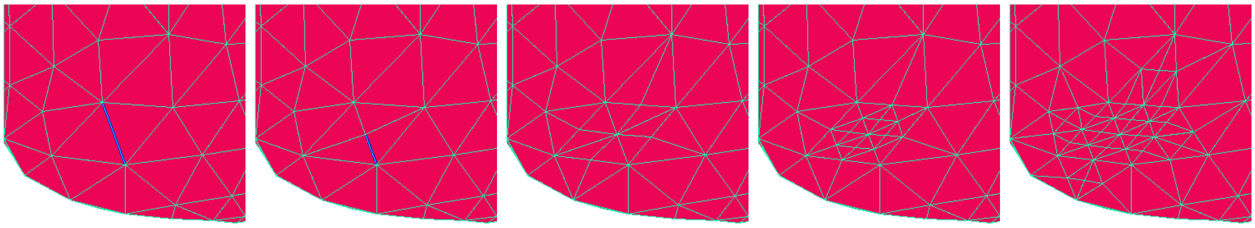


Fig. 4.12 – *Raffinement d'une zone du maillage par divisions successives des arêtes.*

qui seront concernées par le contact. Nous en déduisons l'ensemble des arêtes que nous devons subdiviser. Afin d'éviter la formation d'arêtes trop petites qui pourraient entraîner des instabilités de calcul, nous avons mis en place deux mécanismes de contrôle. Tout d'abord on fixe un seuil sur la taille des arêtes, en dessous duquel on s'interdit toute subdivision. Ensuite, on contrôle le nombre de subdivisions que peut subir une arête. Pour cela, on lui attribue un âge. Les arêtes du maillage initial ont toutes l'âge 0. Dès qu'on divise une arête d'âge n, les deux arêtes résultantes ont l'âge $n + 1$. Lors de la simulation, on pourra alors fixer l'âge maximum que peuvent atteindre les arêtes du maillage.

La figure 4.13 montre le résultat de notre algorithme de raffinement lorsque l'outil entre en contact avec le maillage. Sur la première ligne, l'âge maximum des arêtes est fixé à 1. Lorsque l'instrument touche la surface, toutes les arêtes qui sont dans la zone de contact sont divisées en deux. On a ensuite renouvelé l'expérience en augmentant l'âge à 2, ce qui permet d'obtenir un maillage encore plus fin dans la zone d'intérêt (deuxième ligne de la figure 4.13). On remarque sur le gros plan en bas à droite que seulement deux divisions successives de chaque arête aboutissent déjà à un maillage très fin. Il est, en général, inutile d'aller plus loin.

Nous avons donc couplé cette méthode de raffinement local du maillage avec le processus de découpe. À chaque pas de temps, nous commençons donc par effectuer une première détection de collision entre l'outil et la surface du maillage. Toutes les arêtes qui se trouvent dans la zone de contact sont alors subdivisées (pour de meilleurs résultats, on peut considérer un outil un peu plus large, afin de raffiner une zone un peu plus grande). Nous effectuons ensuite la véritable détection de collision, ainsi que le traitement du contact et des interactions sur le maillage raffiné. Sur l'exemple de la figure 4.14, l'outil a le comportement d'un bistouri à ultrasons, puisqu'il détruit tous les tétraèdres qu'il touche. Sur le première ligne, l'algorithme de raffinement est désactivé. L'outil supprime alors des tétraèdres de grosse taille, ce qui rend le résultat de la découpe peu réaliste. On se rend compte que l'on a supprimé beaucoup trop de matière par rapport à la taille de l'instrument. Sur la seconde ligne, nous autorisons que chaque arête soit subdivisée une fois. L'outil détruit alors des tétraèdres plus petits, ce qui rend le plan de coupe plus régulier et le résultat plus réaliste. On obtient un résultat encore meilleur en autorisant deux subdivisions des arêtes, mais le grand nombre de tétraèdres créés commence à pénaliser la fréquence de simulation.

Cette méthode de raffinement local du maillage par subdivision des arêtes donne donc des résultats intéressants. De plus, on peut envisager des améliorations. En effet, le résultat final du

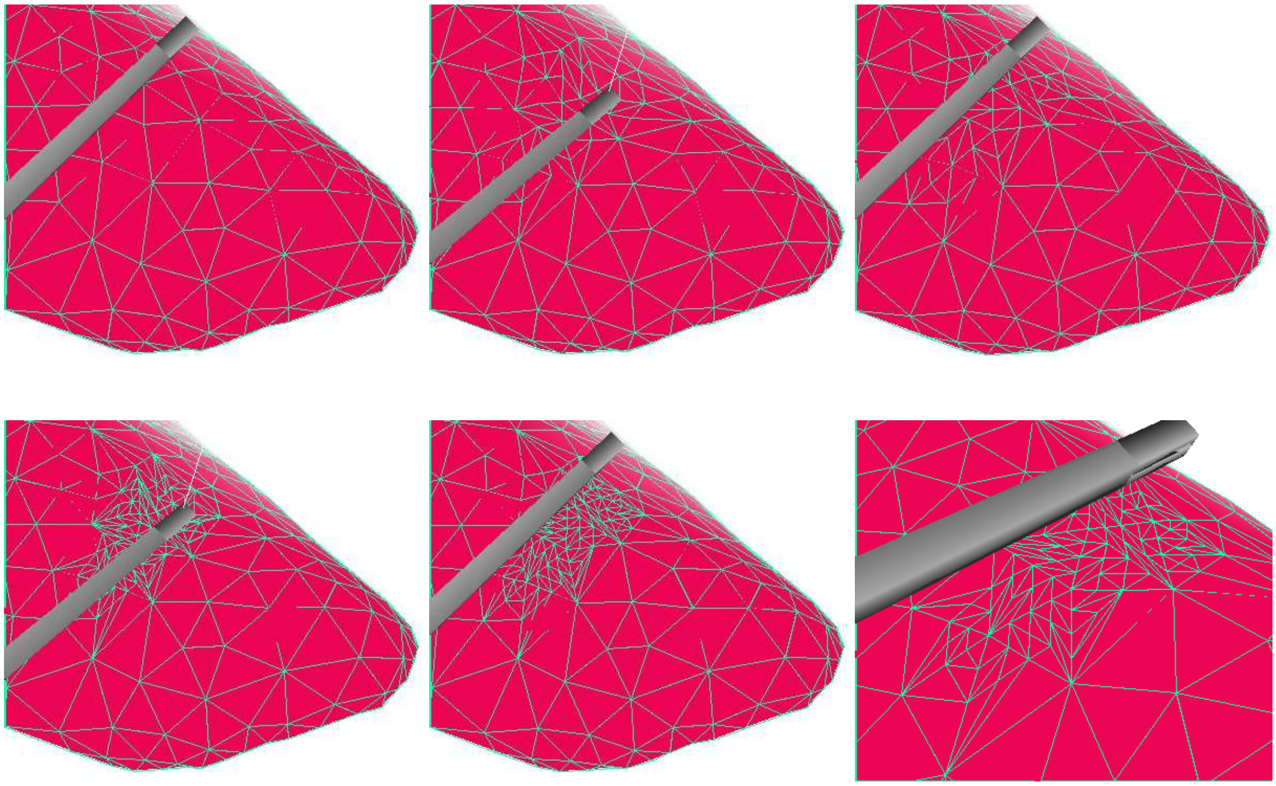


FIG. 4.13 – *Raffinement automatique de la zone du maillage qui est en contact avec l'instrument.*

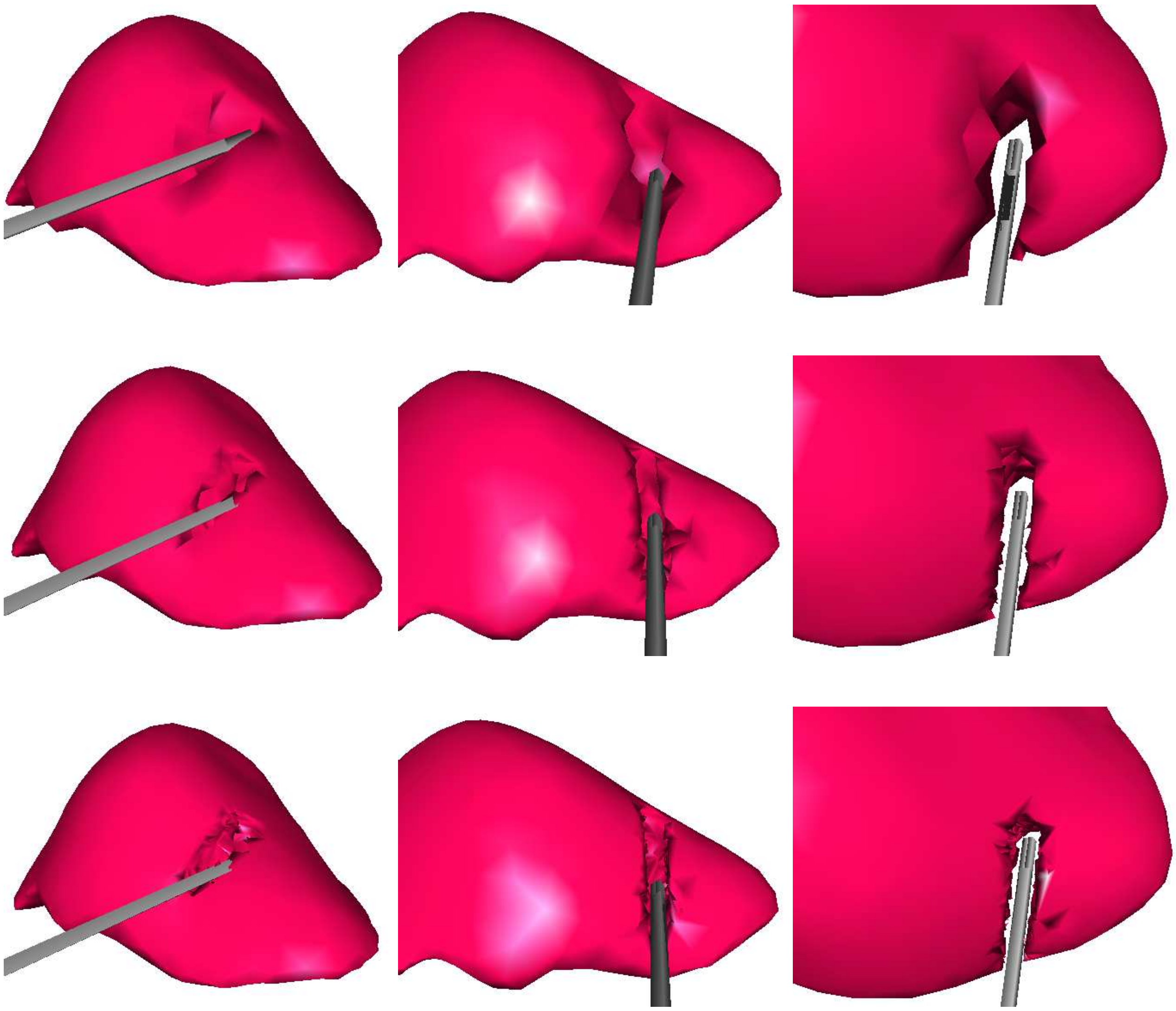


Fig. 4.14 – *Exemple de découpe du foie. La première ligne montre la découpe sur le maillage de départ. Sur la deuxième et la troisième lignes la découpe est couplée avec l'algorithme de raffinement automatique de la zone en contact avec l'outil : chaque arête est subdivisée une fois pour la deuxième ligne et deux fois pour la troisième.*

raffinement dépend de l'ordre dans lequel se fait la subdivision d'un ensemble d'arêtes. Il semble donc possible d'optimiser la qualité du maillage obtenu en triant les arêtes, par exemple en fonction de leur taille. Cependant cette opération ne doit pas être trop coûteuse en temps de calcul.

4.3 Couplage avec un système à retour d'effort

L'utilisation de systèmes à retour d'effort est aujourd'hui largement répandue dans de nombreux domaines. Une grosse partie de la production est bien entendu orientée vers les manettes pour jeux vidéo (microsoft [6], immersion [7]). On peut classer les différents systèmes selon qu'ils sont génériques ou bien orientés vers une application bien précise. Par exemple un "gant haptique" permet de simuler les efforts ressentis sur les doigts et la main lorsqu'on saisit un objet (Dinsmore et al., 1997; Popescu et al., 1999). Il peut être utilisé pour la rééducation de la main et du poignet (Burdea et al., 1997; Popescu et al., 2000). D'autres dispositifs ont un champ d'applications plus important dans la mesure où ils permettent simplement de ressentir une force en un point de l'espace. Le plus connu, et le plus utilisé, est le Phantom [8] (Massie et Salisbury, 1994; Salisbury et Srinivasan, 1997b). La gamme se compose de plusieurs modèles de tailles différentes (voir figure 4.15). Ils possèdent tous 6 degrés de liberté en mouvement (3 translations et 3 rotations) et 3 degrés de liberté en retour d'effort (translations). La gamme s'est récemment enrichie de plusieurs modèles équipés de retour d'effort selon les 6 degrés de liberté (Chen, 1999; Cohen et Chen, 1999).

FIG. 4.15 – *Les différents modèles de "Phantom".*

De nombreux laboratoires travaillant sur la simulation de chirurgie utilisent le Phantom. Suivant les applications, il est utilisé tel quel (Basdogan et al., 1997; Gibson et al., 1998; Burdea et al., 1999), ou bien modifié afin d'adapter ou d'augmenter le nombre de degrés de liberté (Székely et al., 2000b;

6. http://www.microsoft.com/products/hardware/sidewinder/force-feedback/default.htm
7. http://www.immersion.com/entertainment.html
8. http://www.sensable.com/products/phantom.htm

Székely et al., 2000a; Mor, 1998). Enfin, d'autres équipes préfèrent développer leurs propres systèmes à retour d'effort de manière à en maîtriser toutes les caractéristiques et pouvoir les faire évoluer en fonction de leurs besoins (Kühnapfel et al., 1999; Çakmak et Kühnapfel, 2000; Lamy et Chaillou, 1999; Suzuki et al., 1998).

4.3.1 Le "Laparoscopic Impulse Engine"

Pour notre simulateur, nous avons choisi d'utiliser le "Laparoscopic Impulse Engine" développé par la société américaine *Immersion Coorporation*. Ce système à retour d'effort a pour avantage d'avoir été développé spécialement pour la simulation de chirurgie en laparoscopie. Ainsi, il possède une ergonomie parfaite pour cette application : on voit sur la figure 4.16 que la poignée de la pince est la réplique parfaite de celle des véritables instruments de chirurgie (voir figure 1 de l'introduction). De plus on retrouve les 5 degrés de liberté dont dispose le chirurgien : le déplacement dans les trois directions de l'espace (on retrouve le mouvement contraint par le passage par un point fixe), la rotation autour de l'axe de l'instrument et l'ouverture/fermeture de la pince. Les 3 degrés de liberté correspondant aux déplacements sont équipés en retour d'effort via un système de moteurs électriques et de câbles. L'utilisateur dispose d'une translation de 100 mm pour insérer et retirer l'instrument et d'un débattement de 30 degrés par rapport à la verticale. La force maximale qui peut être appliquée sur l'instrument est d'environ 8 N, pour des frottements mécaniques qui n'excèdent pas 0.14 N.

FIG. 4.16 – *Le "Laparoscopic Impulse Engine".*

On observe sur la droite des photos de la figure 4.16 un petit coffret noir. Il contient toute l'électronique nécessaire au fonctionnement de l'instrument, c'est-à-dire en particulier les amplificateurs permettant de commander les trois moteurs du retour d'effort. Ce boîtier est relié à un PC par l'intermédiaire d'une carte contenant les convertisseurs analogique/numérique et numérique/analogique. Ces convertisseurs permettent, dans un sens, de communiquer les informations de position de l'instrument (position, angle et ouverture de la pince) à l'ordinateur et, dans l'autre sens, de recevoir et d'appliquer des forces.

4.3.2 L'interface de communication

Comme c'est souvent le cas lorsqu'on utilise un système à retour d'effort, nos instruments sont gérés par un second ordinateur relié à l'ordinateur principal par un réseau ethernet. Il y a plusieurs raisons à ce choix. La première est historique. L'ordinateur principal doit avoir une puissance graphique conséquente puisqu'il doit être capable d'afficher des scènes complexes, composées d'objets déformables, à une fréquence correspondant au temps réel visuel, c'est-à-dire 25-30 Hz. Pour cela, la première version du simulateur développé au sein du projet Epidaure par Stéphane Cotin (Cotin, 1997) utilisait une station graphique SGI Onyx2 IR. Or, à l'époque, un LIE ne pouvait se raccorder sur un ordinateur que par l'intermédiaire d'une carte au format ISA[9], disponible uniquement sur les PC. Le simulateur était alors constitué de l'Onyx2, relié à un PC gérant le système à retour d'effort. Or l'Onyx2, ainsi que la majorité des stations graphiques performantes, sont extrêmement onéreuses et un simulateur ne fonctionnant que sur ce type de machine semble difficile à implanter en milieu hospitalier en vue d'une évaluation. C'est donc dans un souci de "démocratisation" et de facilité d'installation de notre simulateur, que nous nous sommes dirigés, il y a environ deux ans, vers une solution tout PC. L'idée était alors de n'utiliser qu'un seul PC pour le calcul des déformations et la gestion du système à retour d'effort. Mais un autre problème se pose alors. Le schéma de fonctionnement du simulateur met en œuvre deux boucles asynchrones, une pour l'affichage et l'autre pour le retour d'effort, dont les fréquences de remise à jour sont très différentes (voir paragraphe 4.3.3). On ne peut donc pas les faire fonctionner sur un seul processeur. Or des tests nous ont montrés que même sur un PC bi-processeur fonctionnant sous linux, il était aujourd'hui impossible de faire tourner en parallèle deux boucles dont les fréquences peuvent atteindre 500 Hz (sur un pentium III 500 MHz bi-processeur, on ne peut pas garantir le fonctionnement simultané de deux processus à une fréquence supérieure à 50 Hz). Nous avons donc conservé l'architecture partagée, mais cette fois avec deux PC, un pour la boucle de calcul des déformations et l'affichage, et l'autre pour gérer la boucle de retour d'effort. Ces deux ordinateurs étant reliés par un réseau ethernet, nous avons mis en place un protocole de communication que nous allons décrire maintenant.

4.3.2.1 Description du protocole de communication

Le but de cette interface de communication est de permettre à deux boucles asynchrones d'échanger des informations. Si on suit le cheminement logique de l'information, on obtient le schéma de fonctionnement suivant (voir figure 4.17) :

- Une première boucle lit les informations géométriques du LIE (position, angle, ouverture de la pince) et les envoie en direction de la boucle de simulation. Cet envoi des positions se fait à une fréquence de l'ordre de 300 Hz.
- Sur l'ordinateur supportant la simulation, une boucle reçoit les informations géométriques du LIE et les stocke dans une zone de mémoire partagée, à la même fréquence de 300 Hz.
- La boucle principale de simulation lit la position de l'instrument dans la zone de mémoire

9. *Integrated Systems Architecture*

partagée et, après calcul, renvoie la force exercée sur l'instrument vers l'ordinateur qui gère le retour d'effort. La fréquence de fonctionnement de cette boucle dépend du comportement biomécanique et de la complexité du modèle déformable.

- Enfin, une boucle de lecture récupère alors cette force et la stocke dans une seconde zone de mémoire partagée contenant l'historique des forces appliquées, qui servira à estimer la force qui doit être appliquée sur le système à retour d'effort.
- Le fonctionnement de la boucle d'extrapolation sera détaillé au paragraphe 4.3.3.

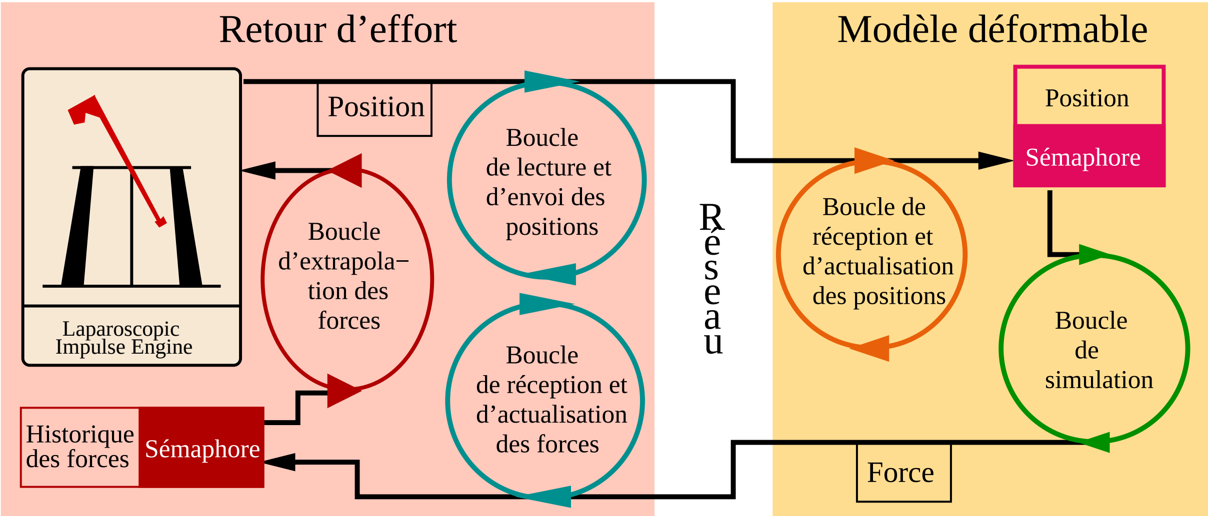


FIG. 4.17 – *Protocole de communication entre la boucle du retour d'effort et la boucle de simulation, via le réseau.*

Certaines caractéristiques de cette interface de communication méritent d'être un peu plus détaillées :

- **L'initialisation :** Les deux parties du simulateur fonctionnant sur deux ordinateurs différents, il est nécessaire de prévoir une phase d'initialisation et de synchronisation des boucles d'envoi et de réception des informations. Lorsque les programmes sont lancés sur chaque ordinateur, ils créent chacun de leur coté les *sockets* de communication et envoient en direction de l'autre un message d'amorçage (*"wake up"*). Le premier qui reçoit ce message passe en mode actif et renvoie à l'autre l'ordre de faire de même. L'échange d'information peut alors commencer. Ce genre de procédure d'initialisation permet de ne pas se soucier de l'ordre dans lequel les programmes sont lancés.
- **Les autres informations échangées :** En plus des messages d'initialisations et les échanges de positions et de forces, d'autres échanges de messages sont nécessaires au bon fonctionnement du simulateur. Tout d'abord, il y a la possibilité d'interrompre un des deux programmes. Le programme qui s'arrête envoie alors à l'autre un message soit pour qu'il se mette en attente d'une nouvelle initialisation, soit pour qu'il s'arrête lui aussi. D'autre part, la boucle de simulation a la possibilité de régler la fréquence d'envoi des positions. Il y a alors un nouvel

échange de messages afin que toutes les boucles se synchronisent à la même fréquence. Enfin, la boucle de simulation avertit le système à retour d'effort au début et à la fin de chaque phase où un outil est effectivement en collision avec le modèle déformable. Ces informations sont nécessaires à la bonne estimation de la force à appliquer sur le LIE.

- **Les zones de mémoire partagées :** Ce sont des espaces de mémoire qui vont être partagés par deux processus indépendants et asynchrones. Les données présentes dans cette zone sont alors surveillées par un *sémaphore*. Ainsi, dès qu'un des processus accède à ces données, simplement pour les lire ou pour les modifier, le sémaphore empêche tout autre accès, et ce, tant que le processus n'a pas "libéré" l'accès. Cela permet en particulier de ne pas lire une information qui est en train d'être modifiée.

4.3.2.2 Performances et perspectives

Malgré l'utilisation du réseau commun à tout le laboratoire, les performances de cette boucle de communication sont satisfaisantes, sauf lorsque qu'un gros transfert de données intervient durant la simulation. Pour pallier ce problème, nous testons le temps mis par les boucles pour lire les messages sur les sockets, et lorsque la lecture est trop longue, la boucle affiche un message à l'écran et supprime le retour d'effort afin de ne pas endommager les LIE.

Un des paramètres importants pour la performance de cette communication est le contrôle de la latence. Par latence, nous entendons le temps écoulé entre l'envoi d'une position de l'instrument et le retour de la force associée à cette position de l'instrument. Cette latence comprend bien évidemment le temps de calcul de la déformation, qui peut aller jusqu'à 0.04s (25 Hz), mais elle doit rester le plus près possible de cette valeur. Dans la pratique, la latence due au transit par le réseau est presque négligeable, puisque pour une fréquence de communication de 300 Hz, elle est comprise entre 0.0035 et 0.01 sec, avec une valeur moyenne autour de 0.004 sec.

Les améliorations futures ne porteront donc pas sur la latence, mais plutôt sur la stabilité de fonctionnement. Afin de ne plus être tributaire du trafic du réseau général du laboratoire, une première solution sera d'installer une seconde liaison ethernet directe entre les deux ordinateurs. Une solution encore plus pratique serait d'avoir un noyau linux "temps réel" installé sur un PC multiprocesseurs, ce qui permettrait de faire fonctionner toutes les boucles au sein d'un même programme, en s'affranchissant du passage au travers du réseau. Enfin, il existe une dernière possibilité, puisque les LIE sont aujourd'hui fournis avec une carte au format PCI[10]. Il est donc possible de les brancher directement sur notre Onyx2 bi-processeur, qui est prévue pour faire fonctionner des boucles asynchrones en temps réel. Mais on revient alors au problème de départ posé par le prix d'une telle machine...

10. *Peripheral Component Interconnect*

4.3.3 Le problème du retour d'effort

La modélisation du toucher et de la sensation de force apportent beaucoup de réalisme dans des applications telles que la simulation de chirurgie. Mais ces sujets comportent encore beaucoup de problèmes non résolus mais dont les débouchés semblent prometteurs (Srinivasan et Basdogan, 1997).

Dans ce paragraphe, nous abordons le problème de la fréquence de simulation nécessaire pour obtenir une bonne sensation de retour d'effort.

4.3.3.1 Description du problème

Nous avons vu que les dispositifs à retour d'effort sont nombreux sur le marché, et qu'ils possèdent des qualités techniques permettant leur utilisation dans de nombreuses applications de simulation. Parallèlement, nous avons présenté des modèles déformables permettant de simuler, en temps réel, des déformations et des découpes réalistes d'objets aux formes géométriques complexes. Mais ces modèles déformables fonctionnent à ce que nous appelons le "temps réel visuel", c'est-à-dire environ 25 Hz. En effet, à cette fréquence d'affichage, la rémanence rétinienne donne une impression de continuité du mouvement et des déformations. Par contre, ces modèles déformables restent trop lents pour être couplés directement à un système à retour d'effort. En effet, le "temps réel haptique", c'est-à-dire la fréquence à partir de laquelle une variation de force nous semble continue, est beaucoup plus élevée, puisque le sens du toucher est beaucoup plus précis que celui de la vue. Les fréquences à atteindre sont comprises entre 300 Hz pour une interaction avec un objet mou et 10 kHz pour modéliser un contact rigide.

Dans la littérature, on trouve deux approches principales permettant d'atteindre le temps réel haptique :

- La première consiste à calculer les forces de manière empirique (Basdogan et al., 1998), par exemple en utilisant une force proportionnelle à la profondeur de pénétration de l'instrument dans l'objet. Le principal problème de cette technique est que la force n'a aucune réalité physique, puisqu'elle n'est pas déduite du calcul de la déformation.
- L'autre possibilité est d'utiliser un modèle physique simplifié. Cette simplification peut se faire dans deux directions, soit en diminuant la taille du maillage représentant l'objet (Burdea et al., 1998), soit en faisant le plus de pré-calculs possible (voir le modèle pré-calculé au paragraphe 2.4.1).

Nous proposons une solution différente, basée sur les caractéristiques du mouvement humain. Il a été montré (Brooks, 1990) que si le sens du toucher est précis (on peut sentir des vibrations jusqu'à 10 kHz, et les variations d'une force entre 30 et 300 Hz), les gestes, en revanche, sont assez lents puisqu'il est possible de les échantillonner de manière convenable autour de 10 Hz. Les forces appliquées sur un système à retour d'effort doivent donc être remises à jour à une cadence élevée. Mais, puisqu'elles sont directement reliées aux mouvements de l'utilisateur, on sait que ces forces vont évoluer lentement. L'idée est donc de se contenter de calculer des forces à une fréquence relativement basse (autour de 30 Hz) et d'estimer la force que l'on va appliquer au système à retour d'effort entre deux calculs successifs de la force.

4.3.3.2 L'extrapolation des forces

Comme nous l'avons décrit au paragraphe 4.3.2, l'architecture de notre prototype de simulateur comprend deux boucles principales (la boucle de calcul de la déformation et la boucle de retour d'effort) qui échangent des informations. Ainsi, la boucle de simulation envoie à la boucle de retour d'effort la force qui doit être exercée sur l'outil. À partir de ces forces fournies à une fréquence de l'ordre de 30 Hz jusqu'à l'instant t_n, nous voulons estimer la force entre l'instant t_n et l'instant t_{n+1} afin de l'appliquer sur l'instrument à une fréquence comprise entre 300 et 500 Hz. Une solution aurait été d'introduire un délai supplémentaire afin d'attendre la valeur de la force à l'instant t_{n+1}, et ensuite d'interpoler les valeurs de la force entre ces deux valeurs (Zhuang et Canny, 1999; Zhuang et Canny, 2000). Mais deux problèmes importants nous ont dissuadés d'utiliser cette méthode. D'abord, le délai entre le mouvement de l'instrument et l'application de la force devient trop important et entraîne une désynchronisation qui présente une gène pour l'utilisateur. Ensuite, rien de garantit que le pas de temps de simulation, et donc l'intervalle entre deux valeurs de la force, soit constant, ce qui complique l'interpolation et nécessite d'augmenter encore le délai. C'est pourquoi nous avons décidé d'estimer la force à tout instant en extrapolant les forces qui viennent d'être calculées par la boucle de simulation.

La boucle de simulation nous fournit donc une série triplets $(t_n, \mathbf{P}_n, \mathbf{F}_n)$, représentant la force $\mathbf{F}_n$ appliquée sur l'instrument à l'instant t_n, lorsqu'il se trouve à la position $\mathbf{P}_n$. L'intervalle de temps séparant deux triplets est de l'ordre de 0.04 s (temps réel visuel) et n'est pas nécessairement constant. Nous devons donc choisir une fonction d'extrapolation $\mathbf{F}(t)$ qui fournira une estimation de la force appliquée sur l'instrument à tout instant t $(t_n \leq t < t_{n+1})$, à partir des données connues jusqu'à l'instant t_n : $\{(t_i, \mathbf{P}_i, \mathbf{F}_i),\ i = 0..n\}$.

Une méthode très populaire pour ce genre d'estimation est le filtrage de Kalman. En effet, cette méthode est définie dans la préface de (Chui et Chen, 1987), comme un processus d'estimation d'un état optimal, qui donne un algorithme linéaire, non biaisé et robuste pour estimer un état inconnu d'un système dynamique à partir de données discrètes bruitées acquises en temps réel. Mais dans notre cas, l'absence de perturbations aléatoires nous a conduit vers des modèles d'extrapolation relativement simples.

Afin de valider notre approche, nous avons testé plusieurs méthodes d'extrapolation, que nous allons décrire maintenant.

Extrapolation constante

La manière la plus simple d'estimer la valeur d'une donnée est de la considérer égale à sa dernière valeur connue :

$$\mathbf{F}^{cte}(t) = \mathbf{F}_n$$

$$t_n \leq t < t_{n_1}$$

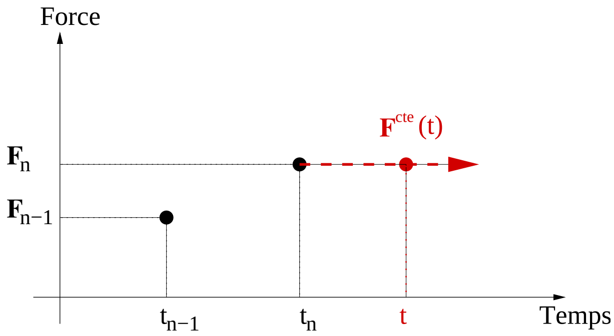


FIG. 4.18 – *Extrapolation constante.*

Cette méthode d'extrapolation a plusieurs avantages. Tout d'abord, elle ne nécessite aucun calcul. De plus, comme toutes les forces appliquées résultent du calcul de la déformation, on est sûr d'appliquer uniquement des forces valides. Il n'y a aucun risque d'appliquer une force erronée qui, si elle est trop grande, risquerait d'endommager le LIE. Enfin, cette méthode correspond au fonctionnement de base du LIE. Par contre, elle donne une estimation qui est fausse dès que la force varie d'un instant à l'autre. Son principal défaut est l'aspect discontinu (en marches d'escalier) des forces estimées, qui donne la sensation de toucher une surface rugueuse.

Extrapolation linéaire en fonction du temps

Une autre façon d'estimer la valeur courante d'un signal qui varie en fonction du temps est de l'extrapoler linéairement en fonction du temps. Cela consiste à considérer que le signal va continuer à varier dans les mêmes proportions qu'entre les deux instants précédents :

$$\mathbf{F}^{tps}(t) = \mathbf{F}_n + \frac{t - t_n}{t_n - t_{n-1}}(\mathbf{F}_n - \mathbf{F}_{n-1})$$

$$t_n \leq t < t_{n_1}$$

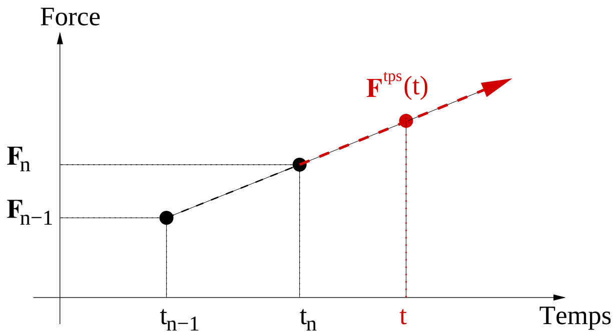


FIG. 4.19 – *Extrapolation linéaire en fonction du temps.*

Cette méthode donne des résultats bien meilleurs que la première. En particulier, on ressent nettement moins de discontinuités. Par contre, on doit faire face à un nouveau problème. Comme les forces appliquées n'ont pas été directement calculées par la boucle de simulation, elle peuvent

être arbitrairement grandes. Ces pics de forces, qui sont des erreurs, se produisent notamment lorsque le pas de temps augmente de manière significative.
Nous avons donc essayé de trouver une méthode plus robuste.

Extrapolation linéaire en fonction de la position

L'idée est d'utiliser une information plus pertinente que le temps pour estimer la force. Dans la boucle de simulation, le déformation du modèle est due à son contact avec l'outil de chirurgie. Et donc la force qu'exerce en retour le modèle déformable sur l'instrument est directement reliée à la position de l'instrument. Or on peut connaître la position de l'outil aussi souvent qu'on le désire (la bande passante du LIE en lecture est supérieure à 1000 Hz). Nous allons donc estimer la force non pas en fonction du temps qui s'est écoulé depuis la donnée de la dernière force, mais en fonction du déplacement subi par l'instrument.
Le raisonnement est le suivant :

- À partir des positions $\mathbf{P}_{n-1}$ et $\mathbf{P}_n$ de l'instrument aux instants t_{n-1} et t_n, on construit une droite qui approxime localement le mouvement de l'instrument.
- On projette alors la position courante $\mathbf{P}$ de l'instrument sur cette droite pour obtenir le point $\mathbf{P}'$, qui est une estimation de la position qu'aurait eu l'instrument s'il avait continué selon la trajectoire définie par les deux instants précédents.
- On utilise alors les rapports des longueurs séparant les points $\mathbf{P}_{n-1}$, $\mathbf{P}_n$ et $\mathbf{P}'$ pour estimer la force qu'il faut appliquer au système à retour d'effort.

$$\mathbf{F}^{pos}(t) = \mathbf{F}_n + \frac{\|\mathbf{P}' - \mathbf{P}_n\|}{\|\mathbf{P}_n - \mathbf{P}_{n-1}\|}(\mathbf{F}_n - \mathbf{F}_{n-1})$$

$$t_n \leq t < t_{n_1}$$

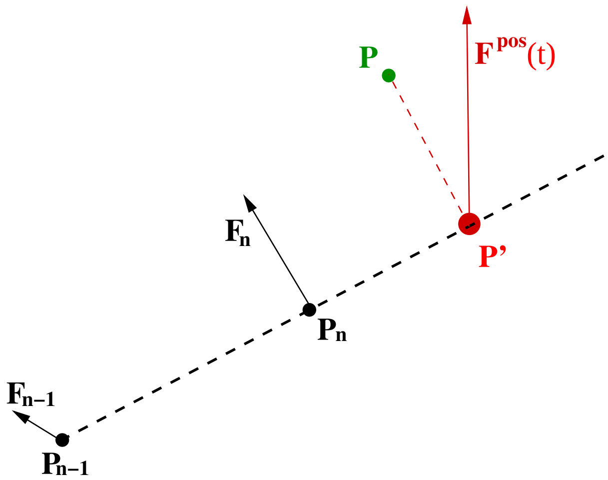


FIG. 4.20 – *Extrapolation linéaire en fonction de la position.*

Nous avons intégré ces trois méthodes d'extrapolation à notre simulateur de chirurgie. Afin de les évaluer et de les comparer, nous avons réalisé plusieurs expériences, dont nous allons maintenant donner les résultats.

4.3.3.3 Résultats

La première expérience que nous avons réalisée a été de se placer dans les conditions de simulation, et d'activer tour à tour les différentes méthodes d'extrapolation, afin d'effectuer une évaluation qualitative basée sur le réalisme du retour d'effort obtenu. Comme on peut s'y attendre, l'extrapolation constante donne une sensation de granularité de la force qui diminue nettement le confort d'utilisation. L'extrapolation linéaire en fonction du temps apporte une nette amélioration en donnant une sensation de continuité très réaliste. Cependant, quelques pics de force interviennent de temps en temps, en particulier si le pas de temps varie de manière importante. L'extrapolation linéaire en fonction de la position améliore encore la sensation de continuité de la force en diminuant notablement les pics de force dus aux erreurs d'estimation. On peut expliquer cette amélioration de la manière suivante. Pour l'extrapolation en fonction du temps, on fait l'hypothèse que la force varie linéairement en fonction du temps. Si l'on admet que la force est reliée au mouvement de l'instrument, cela veut dire que l'on fait l'hypothèse d'un mouvement de l'instrument linéaire et surtout uniforme. Par contre dans le cas de l'extrapolation en fonction de la position, on se limite à un mouvement qui est certes linéaire, mais plus uniforme puisque le temps n'entre plus en jeu. La différence entre les deux méthodes linéaires est donc que l'une permet une variation de la vitesse de l'instrument, alors que l'autre fait l'hypothèse d'une vitesse constante en fonction du temps. L'hypothèse de base de l'extrapolation linéaire en fonction de la position étant moins contraignante, il est normal qu'elle engendre moins d'erreurs.

Cependant, cette évaluation basée sur la sensation n'est pas parfaitement objective. Aussi, nous avons mis en place une série d'expériences permettant une comparaison plus quantitative des différentes méthodes, ainsi qu'un calcul des erreurs d'extrapolation commises. Pour cela nous avons enregistré, durant des sessions de simulation, la position de l'instrument, la force appliquée et le temps en sortie de la boucle de simulation. Nous obtenons ainsi un échantillon de simulation que nous allons pouvoir soumettre à nos méthodes d'extrapolation. Nous avons utilisé un modèle déformable relativement simple, afin de faire une acquisition des données à une fréquence de 80 Hz. Nous avons pu ainsi sous-échantillonner ces données afin d'en choisir la fréquence. Ces données ont été interpolées à posteriori. Cette interpolation servira de référence pour le calcul des erreurs. La figure 4.21 présente un exemple d'échantillons de simulation sous-échantillonnés 4 fois (pour obtenir la fréquence de 20 Hz). Afin de faciliter la représentation, nous ne considérons les forces que dans le plan (y,z) en fonction du temps. Les lignes représentent les échantillons enregistrés et le trait gras l'interpolation de ces échantillons. Sur la gauche de la figure, nous proposons une décomposition polaire de la force (norme et angle par rapport à l'axe des z) qui sera utilisé dans le reste du paragraphe.

Nous avons donc soumis l'enregistrement sous-échantillonné à 20 Hz à nos trois méthodes d'extrapolation. Les résultats obtenus sont présentés, sous forme polaire, sur la colonne de gauche de la figure 4.22. L'erreur, c'est-à-dire la différence entre la force extrapolée et la force interpolée est

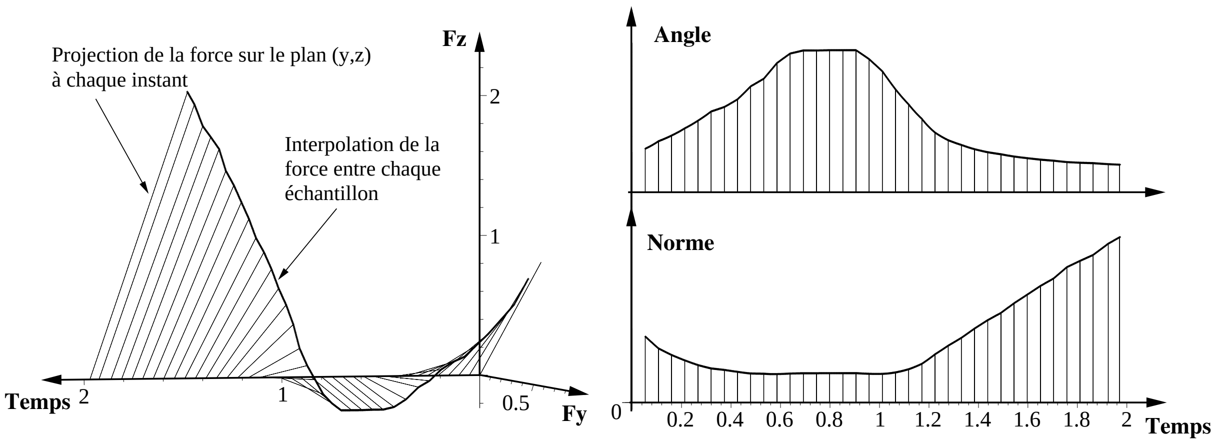


FIG. 4.21 – *Représentation des données enregistrées en sortie de la boucle de simulation (20 Hz), et de leur interpolation a posteriori (500 Hz).*

représentée, elle aussi en polaire sur la colonne de droite. Ces résultats montrent assez nettement la supériorité de l'extrapolation linéaire en fonction de la position.

Nous avons réalisé de nombreuses expériences comparables, en faisant varier la fréquence de simulation et la vitesse de l'instrument. C'est toujours l'extrapolation en fonction de la position qui a donné les meilleurs résultats. Mais nous avons pu en plus faire ressortir l'importance du rapport entre la vitesse de l'instrument et la fréquence de simulation. En effet, ce rapport conditionne l'échantillonnage spatial du mouvement de l'instrument, c'est-à-dire la distance parcourue par l'instrument entre deux échantillons. Si cette distance est trop grande, l'hypothèse d'un mouvement linéaire entre chaque échantillon ne sera pas vérifiée, et la méthode fonctionnera mal. En observant des enregistrements d'opérations en laparoscopie et en reproduisant des gestes similaires sur notre simulateur, nous obtenons des vitesses du bout de l'instrument de l'ordre du centimètre par seconde. Or nos tests nous ont montré qu'un échantillonnage spatial du mouvement de l'instrument de l'ordre de 0.5 mm garantit une bonne qualité de l'extrapolation. Or un tel échantillonnage correspond dans l'exemple décrit par le tableau 4.1, à une fréquence de simulation de 33 Hz pour une vitesse moyenne de l'instrument de $1.7 cm/s$.

De plus, nous avons vérifié, sur différents exemples, que les erreurs maximales de l'extrapolation linéaire en fonction de la position intervenaient systématiquement à des moments où la vitesse de l'instrument était élevée c'est-à-dire, pour une fréquence donnée, à un moment où l'échantillonnage du mouvement devenait insuffisant.

Afin de comprendre parfaitement d'où venaient les erreurs les plus importantes, nous avons fait des enregistrements avec des mouvements de l'instrument rapides et saccadés. L'exemple de la figure 4.23 regroupe les différentes causes d'erreurs que nous avons identifiés.

En ①, on observe une forte variation de l'angle de la force. En fait, cet instant correspond à une inversion du sens de la force, puisque la norme passe presque par zéro. L'erreur induite par

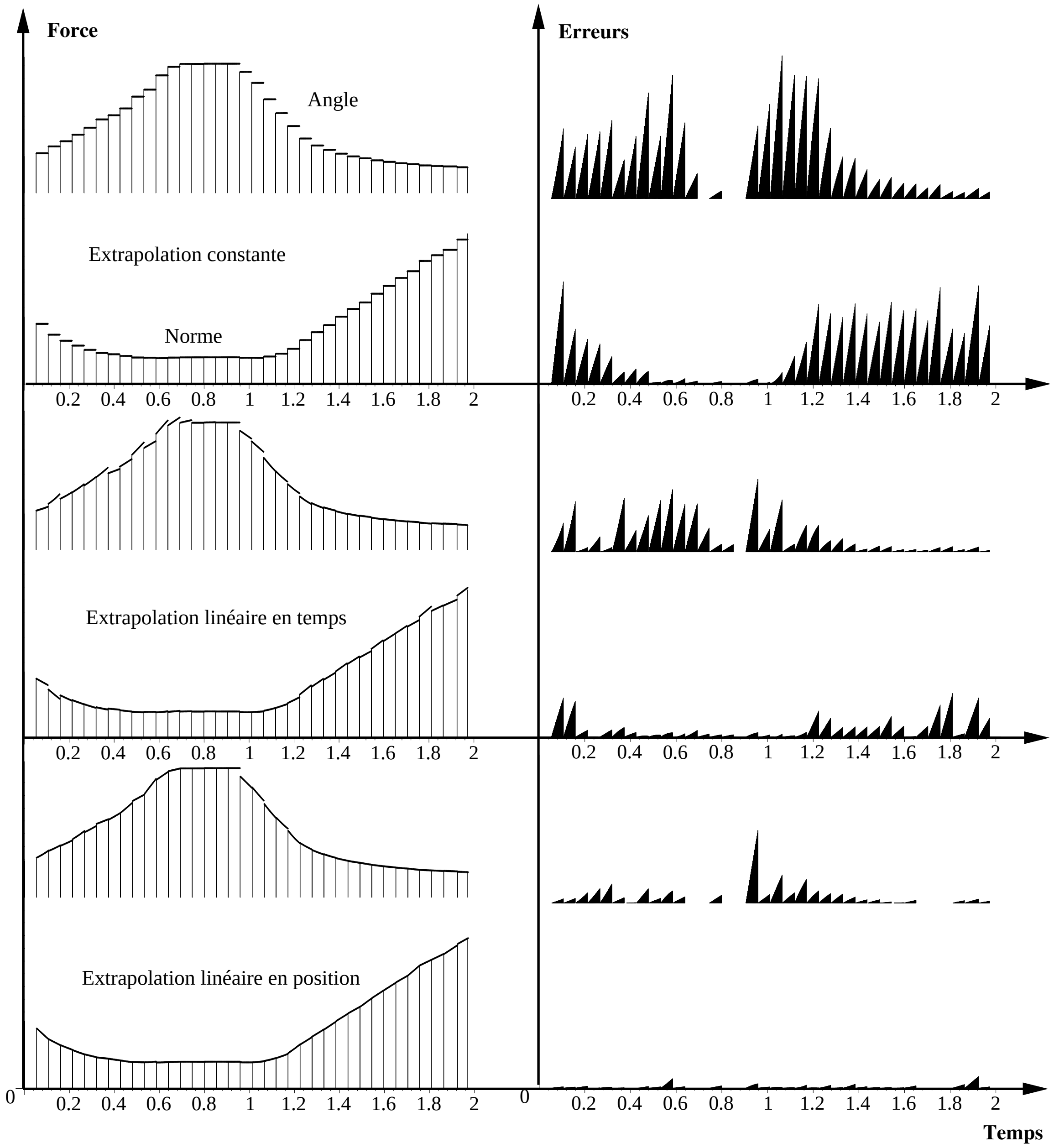


FIG. 4.22 – *Comparaison des trois méthodes d'extrapolation et de leurs erreurs respectives.*

Vitesse moyenne	$0.017\ ms^{-1}$		
Vitesse max.	$0.078\ ms^{-1}$		
Fréquence	$33\ Hz$		
Échantillonnage spatial	$0.5\ mm$		
méthode	erreur moyenne	erreur maximale	force maximale
constante	1.1 %	56 %	133 %
linéaire en temps	0.3 %	9 %	109 %
linéaire en position	0.1 %	7 %	106 %

TAB. 4.1 – *Calcul des erreurs liées à l'extrapolation lors d'une session de simulation*

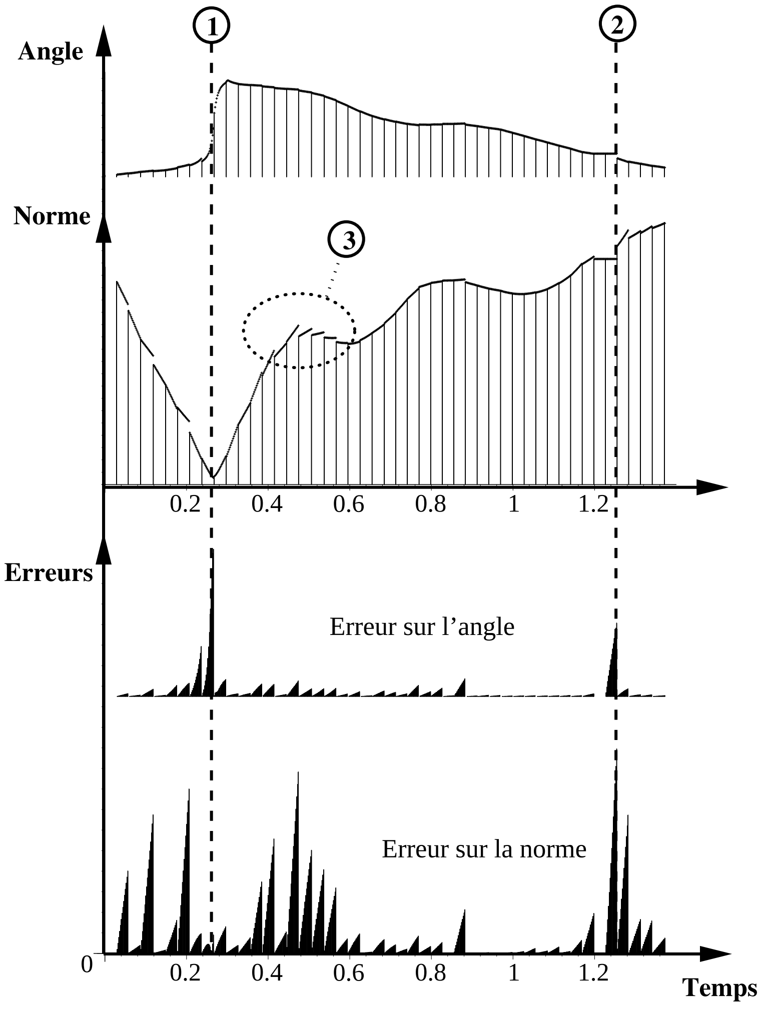


FIG. 4.23 – *Interprétation des différentes sources d'erreurs importantes.*

cette forte variation de la direction de la force n'est pas importante, puisqu'elle intervient à un instant de la simulation où la norme est très faible. On ne la ressentira absolument pas. L'erreur ②, en revanche, est nettement perceptible. Entre les deux instants précédant cette erreur, la force et surtout la position n'ont pas changées. L'extrapolation ne possède alors plus d'informations pour estimer les forces à venir. Ce problème se pose dès que l'instrument s'immobilise, ce qui est quand même relativement rare car l'instrument est dirigé par l'utilisateur qui peut difficilement le maintenir parfaitement immobile. Nous traitons ce cas particulier simplement en ne tenant pas compte des échantillons pour lesquels la position n'a pas changé par rapport à l'instant précédent.

Par contre, l'erreur ③ n'est pas contrôlable, car elle correspond simplement à une vitesse trop élevée de l'instrument. C'est donc bien le point clé pour contrôler les erreurs de la méthode d'extrapolation linéaire en fonction de la position. Pour une fréquence de simulation donnée, on ne doit pas dépasser une certaine vitesse de l'instrument, afin de préserver un échantillonnage spatial du mouvement suffisant (c'est-à-dire, d'après nos expériences, inférieur à 0.5 mm).

4.4 Conclusions

Dans ce chapitre, nous avons présenté différents éléments qu'il faut développer autour d'un modèle déformable pour construire un véritable simulateur de chirurgie laparoscopique sur le foie. Nous avons commencé par définir les outils virtuels avec lesquels l'utilisateur va effectuer des gestes chirurgicaux sur le modèle de foie. Nous avons choisi ensuite une méthode de détection de collision particulièrement efficace que nous avons mis à profit afin de modéliser très précisément le contact entre un instrument et la surface du foie. Nous avons proposé un algorithme de traitement de la collision tenant compte des comportements différents que peuvent avoir le manche et l'extrémité de l'instrument. De plus, nous avons modélisé plusieurs gestes de base et nous nous sommes particulièrement intéressés au problème de la découpe, pour lequel nous apportons une méthode de raffinement local du maillage qui améliore nettement le réalisme.

Nous avons ensuite abordé le problème de l'utilisation d'un système à retour d'effort. L'architecture de notre prototype de simulateur étant distribuée sur deux ordinateurs, un pour le calcul de la déformation et l'affichage, l'autre pour la gestion de l'instrument à retour d'effort, nous avons été amenés à développer une interface de communication. Un protocole adapté permet aux programmes fonctionnant sur chacun des ordinateurs de se synchroniser et d'échanger des informations de force et de position avec un minimum de latence et un contrôle des performances.

Enfin, nous proposons une méthode permettant de réconcilier les temps de calcul des modèles déformables physiques réalistes avec les fréquences élevées de remise à jour des forces nécessaires pour l'utilisation de systèmes à retour d'effort. Cette méthode tient compte de la sensibilité du sens du toucher, ainsi que des caractéristiques des mouvements humains. Elle est fondée sur une boucle fonctionnant à haute fréquence (>300 Hz) qui extrapole les forces calculées par la boucle de simulation qui fonctionne à une fréquence de 30 Hz. Nous avons testé trois fonctions d'extrapolation : l'extrapolation constante, l'extrapolation linéaire en fonction du temps, et une fonction plus originale d'extrapolation linéaire en fonction de la position, qui met à profit la possibilité de connaître la position de l'instrument à n'importe quel instant. De nombreux tests ont montré que cette dernière méthode donne de très bons résultats dès lors que l'échantillonnage spatial du mouvement de l'instrument est suffisamment fin, c'est-à-dire pour une fréquence de simulation donnée, que la vitesse de l'instrument n'est pas trop grande. Cela nous a permis de conclure que pour la vitesse des gestes des chirurgiens, qui est de l'ordre de quelques centimètres par secondes, une boucle de simulation fonctionnant en temps réel visuel (environ 25 Hz) permet de produire un retour de bonne qualité grâce à l'extrapolation linéaire en fonction de la position.

Chapitre 5

Simulation de chirurgie hépatique avec retour d'effort : schéma d'élaboration

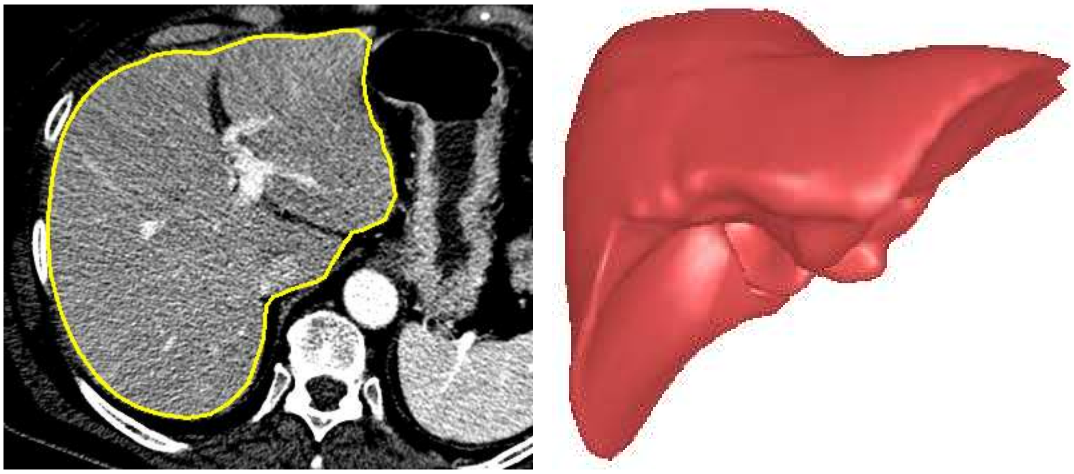
Après avoir présenté, dans les chapitres précédents, les différents travaux que nous avons effectués autour du thème de la simulation de chirurgie, nous nous attardons un peu sur le processus global d'élaboration, qui va de la construction du maillage jusqu'à la simulation de quelques gestes chirurgicaux.

5.1 Introduction

En dehors des différents développements que nous avons exposés tout au long de ce manuscrit, la mise au point d'un scénario de simulation comprend un certain nombre d'étapes que nous avons jusque-là laissées dans l'ombre. Afin de réparer cette injustice, nous consacrons la première partie de ce chapitre aux aspects pratiques et techniques nécessaires à la construction d'un maillage de l'organe qui nous intéresse, et au choix des coefficients d'élasticité et des conditions aux bords. La seconde moitié de ce chapitre propose une description un peu plus détaillée de quelques "modules de simulation".

5.2 Fabrication du maillage

Le projet Epidaure possède une grosse expérience du traitement des images médicales. En particulier, de nombreux travaux ont été menés sur la segmentation des images, c'est-à-dire la délimitation des structures anatomiques d'intérêt. Ainsi J. Montagnat[1] a développé durant sa thèse (Montagnat, 1999) un modèle de surface déformable utilisant des informations extraites de l'image, des contraintes de régularité et une notion de forme *a priori*, pour venir délimiter les contours de l'organe choisi. Cette méthode permet, entre autres, de reconstruire la surface du foie à partir d'images scanner X (figure 5.1).



(a) Trace de la surface déformable dans une des coupes du scanner.

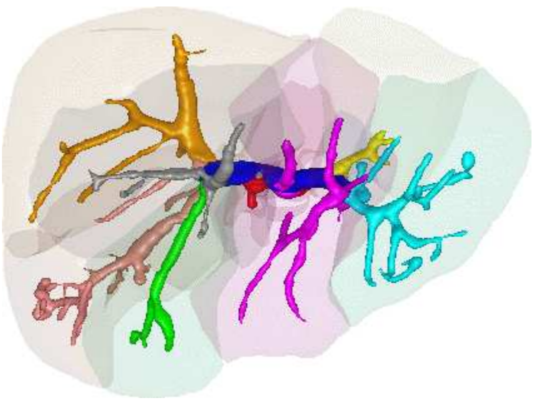
(b) Surface segmentée du foie.

FIG. 5.1 – *Extraction de l'enveloppe externe du foie à partir d'un scanner (Montagnat, 1999).*

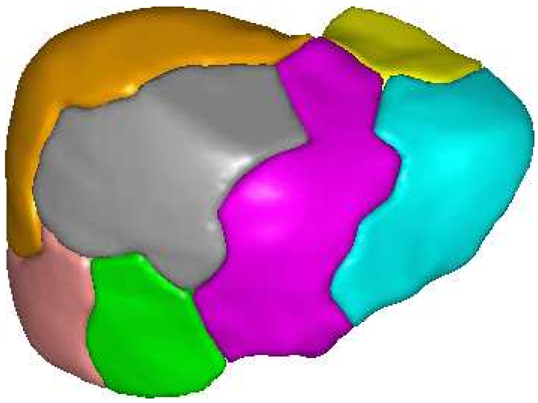
Il est ensuite possible de reconstruire d'autres structures intéressantes à l'intérieur du foie. C'est par exemple la cas des vaisseaux sanguins et des éventuelles tumeurs ou lésions. Les vaisseaux

1. J. Montagnat : http://www-sop.inria.fr/epidaure/personnel/Johan.Montagnat/montagnat.html

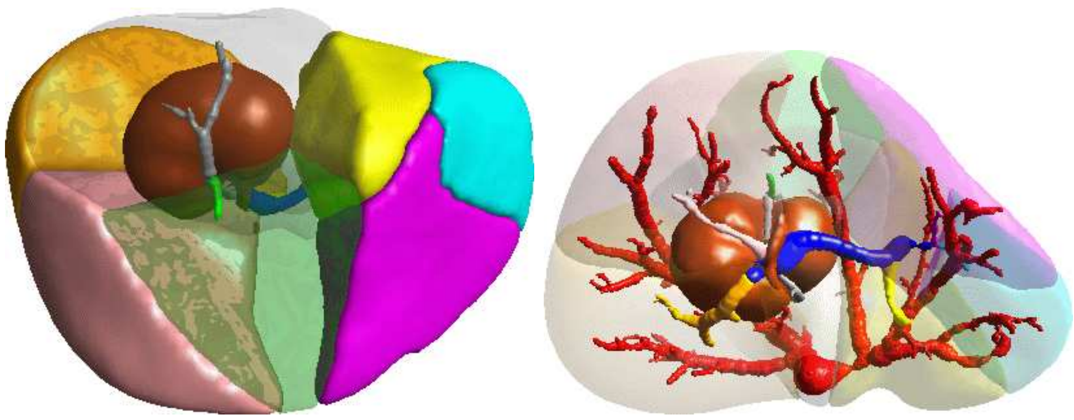
sanguins, et en particulier le réseau porte, ont une grande importance anatomique, puisqu'ils vont conditionner la subdivision du foie en plusieurs segments. Un segment représente une zone du foie qui est irriguée par la même branche du réseau hépatique. Cela signifie qu'en cas d'opération chirurgicale visant à supprimer une partie du foie, il faut impérativement supprimer le ou les segments concernés dans leur intégralité. L. Soler[2] a travaillé, durant sa thèse (Soler, 1998), sur la décomposition automatique du foie en segments à partir de la reconstruction de l'arbre portal (réseau de ramifications de la veine porte à l'intérieur du foie). Sa méthode permet aussi de détecter les structures pathologiques et ainsi de proposer au chirurgien un outil de planification préopératoire qui lui indique quels sont les segments qui doivent êtres retirés (voir figure 5.2).



(a) Segmentation de la veine porte et de ses ramifications.



(b) Décomposition du foie en segments anatomiques.



(c) Segmentation d'une lésion qui peut alors être positionnée par rapport aux segments et aux vaisseaux du foie.

FIG. 5.2 – *Segmentation des vaisseaux et des éventuelles lésions en vue d'une aide à la planification chirurgicale (Soler, 1998).*

2. L. Soler : http://www.virtual-surg.com/equipe/soler.htm

Cette segmentation nous fournit donc la surface du foie ou les segments sous la forme d'une triangulation. Mais ces maillages très précis comportent souvent un grand nombre de triangle. Nous devons donc commencer par les décimer. Nous utilisons pour cela un mailleur développé par le projet Gamma[3] de l'INRIA Rocquencourt. Ce mailleur surfacique, qui se nomme YAMS (Borouchaki et Frey, 1999), permet de modifier le nombre de triangles utilisés pour modéliser une surface et d'optimiser la position des sommets afin de rendre la triangulation la plus homogène possible. L'étape suivante consiste à construire une tétraédrisation à partir de cette surface. Pour cela, nous utilisons les mailleurs volumiques développés eux aussi dans le projet GAMMA, GHS3D (George, 1997; George, 1999b) et GAMHIC3D (George et Borouchahi, 1999; George, 1999a) (GHS3D est distribué par la société **SIMULOG**[4] sous le nom TetMesh[5]). Ces mailleurs permettent de remplir avec des tétraèdres l'intérieur d'une surface fermée. Lors de cette étape, il est aussi possible de maîtriser la résolution du maillage final.

5.3 Définition du comportement biomécanique

En plus du maillage géométrique qui représente le foie, nous avons besoin de deux informations supplémentaires pour faire fonctionner correctement nos modèles déformables. La première est une information rhéologique, puisqu'il faut donner aux coefficients de Lamé des valeurs réalistes. La seconde est anatomique, car nous devons connaître les zones du foie qui peuvent être considérées comme fixées.

5.3.1 Coefficients d'élasticité : rhéologie

Les coefficients de Lamé s'expriment en fonction du module de Young et du rapport de Poisson (équation 1.27). Le rapport de Poisson caractérise l'incompressibilité de matériau. Il faudra donc le choisir le plus proche possible de 0.5, tout en prenant garde de ne pas rendre le schéma de résolution trop instable. Par contre, le module de Young caractérise réellement l'élasticité du matériau. Malheureusement, il est très difficile à mesurer sur un organe tel que le foie, car il faut le faire *in vivo*. En effet, le foie étant rempli de sang, il n'aura plus le même comportement *ex-vivo*, une partie du sang s'étant écoulée ou bien ayant coagulé.

De nombreux travaux sont menés sur la mesure des caractéristiques biomécaniques d'organes, sur des humains ou sur des animaux (Davies et al., 1999; Maaß et Kühnapfel, 1999).

Une méthode intéressante est proposée par Vuskovic *et al.* pour déterminer de manière non invasive les valeurs des coefficients élastiques utilisés dans les lois de comportement (Vuskovic et al., 1999b; Vuskovic et al., 1999a; Vuskovic et al., 2000; Kauer et al., 1999). Ils ont mis au point un instrument, qui se présente sous la forme d'un tube prolongé d'une poignée, qui peut être utilisé aussi bien en chirurgie ouverte, en laparoscopie, ou lors d'expériences *ex-vivo*. Le principe est de

3. Gamma : http://www-rocq.inria.fr/gamma/fra.htm
4. SIMULOG : http://www.simulog.fr/
5. TetMesh : http://www.simulog.net/tetmesh/

placer l'extrémité du tube sur la surface de l'organe, et d'y exercer un force de succion. Une partie de l'organe est alors aspirée vers l'intérieur du tube et une caméra enregistre le profil de la déformation. Ce profil est par la suite reconstruit. Des modèles déformables, basés sur les lois élastiques Neo-Hookéenne et de Veronda-Westmann (Veronda et Westmann, 1970; Fung, 1993), sont ensuite utilisisés pour simuler la même expérience. Les coefficients de ces modèles sont alors ajustés de manière à retrouver exactement la même déformation. Malheureusement, les dernières publications ne font état que d'expériences préliminaires réalisées sur le rein. Ils obtiennent cependant des valeurs du module de Young entre 500 et 800 Pa, ce qui correspond assez bien aux valeurs proposées par d'autres équipes.

Dans le cadre des projets AISIM et CAESARE que nous avons présentés dans le chapitre d'introduction, M. Thiriet a regroupé, sur un très intéressant cite internet consacré à la rhéologie du foie[6], les résultats de travaux menés par trois équipes différentes. D. Dan a étudié, durant sa thèse, le comportement du foie dans des conditions de chocs (Dan et Flaud, 1998; Dan, 1999; Dan et al., 2000). Ses expériences ont été réalisées *in vitro*, sur des échantillons transfusés. Elle a ainsi pu déterminer des valeurs du module de Young allant de 500 à 600 Pa pour une plage de déformations comprises entre 0 et 20 % de la taille des échantillons. Mais ces expériences furent menées en séparant le parenchyme de la capsule de Glisson. Il est donc difficile d'en déduire une valeur pour le foie complet. Une autre étude a été menée par F. Carter, cette fois *in vivo*, à l'aide d'un instrument équipé de capteurs de position et de force (Carter et Davies, 1997; Carter, 1998). Le foie est alors considéré dans sa globalité. Elle obtient alors un module d'élasticité de l'ordre de 10^5 Pa. Une dernière étude menée par K. Miller[7] (Miller, 2000), à partir d'expériences réalisées par Melvin *et al.* sur des foies de macaques isolés mais perfusés (vascularisation intacte) (Melvin et al., 1973), donne des valeurs de l'ordre de $6 * 10^3$ Pa.

On se rend bien compte de la variabilité des résultats, suivant les conditions de l'étude et les dispositifs expérimentaux. Il est donc très difficile de choisir une valeur en étant sûr qu'elle soit réaliste. Pour nos simulations, nous considérons que la valeur du module de Young est de l'ordre de $10^4 - 10^5$ Pa.

5.3.2 Contraintes aux bords : anatomie

Afin de donner un aspect réaliste aux déformations de nos organes, nous devons respecter les conditions d'attachement, c'est-à-dire les zones par lesquelles le modèle est fixé. Comme la plupart des organes abdominaux, le foie n'est pas fixé à une partie du squelette. Il est simplement tenu en place par les organes qui l'entourent, à savoir la paroi abdominale, les poumons, l'estomac et les intestins (voir figure 5.3). Par contre, lors d'une opération en laparoscopie, l'abdomen du patient est gonflé avec du gaz, afin de laisser de la place aux chirurgiens pour opérer, et le foie est un peu dégagé des organes qui l'entourent. Dans ces conditions, la seule zone qui peut être considérée

6. http://www-rocq.inria.fr/Marc.Thiriet/Glosr/Bio/Foie/RheoFoie.html
7. K. Miller : http://www.mech.uwa.edu.au/kmiller/

comme fixe se trouve sur la face antérieure, et correspond au hile hépatique, c'est-à-dire l'endroit où la veine cave et l'artère hépatique entrent dans le foie et où les veines sus-hépatiques en repartent (voir partie droite de la figure 5.3).

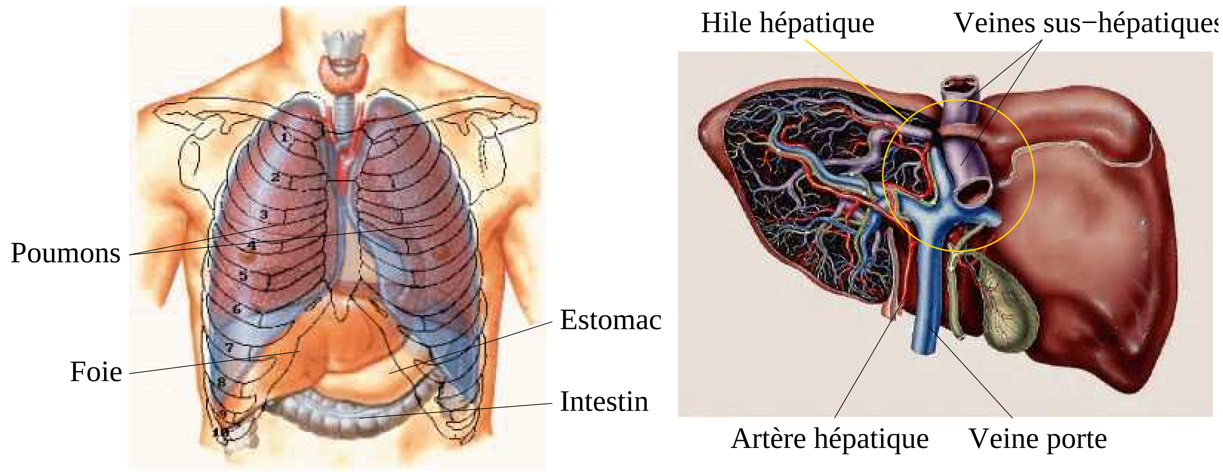


FIG. 5.3 – *Quelques éléments d'anatomie. Positionnement du foie par rapport aux autres organes de l'abdomen (à gauche) et détails du point d'attachement du foie au niveau du hile hépatique (à droite).*

C'est donc à cet endroit que nous avons fixé les sommets de notre maillage de foie, comme on peut le voir sur la figure 5.4, où les sommets marqués sont contraints à un déplacement nul.

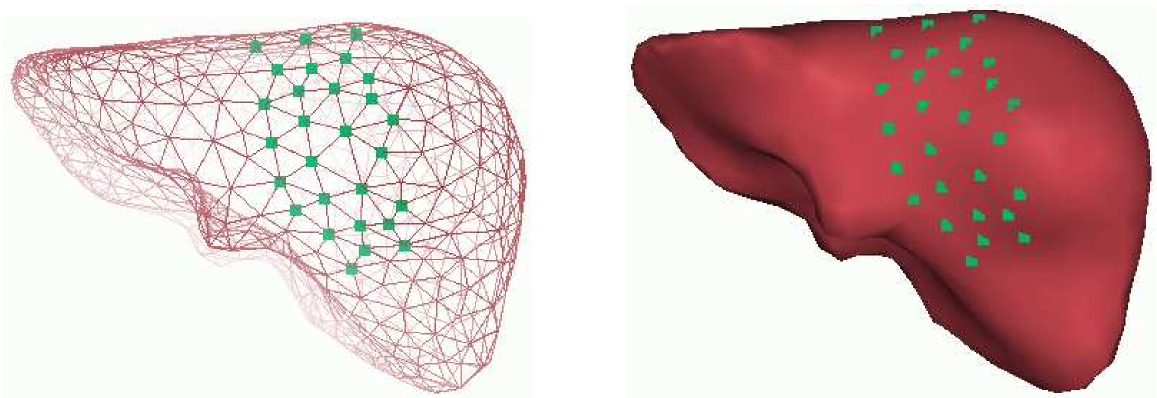


FIG. 5.4 – *Sommets fixés sur la partie antérieure du foie correspondant au hile hépatique.*

! Remarque : La plupart de ces informations proviennent du site internet développé par M. Thiriet, consacré au foie [8]. Nous encourageons vivement toute personne s'intéressant à cet organe d'aller le visiter.

8. http://www-rocq.inria.fr/Marc.Thiriet/Glosr/Bio/Foie/index.html

5.3.3 Problèmes de validation

Un des problèmes inhérents aux outils informatiques appliqués au milieu médical, est celui de la validation. C'est encore plus vrai dans le cas de la simulation de chirurgie. Il n'existe, en effet, aucun moyen simple de vérifier que la déformation que l'on va simuler est proche de ce qui se passerait dans la réalité. Il faudrait, pour cela, être en mesure de reconstruire la déformation subie par un organe à l'intérieur de l'abdomen du patient, pour ensuite la comparer à ce qui est obtenu avec le modèle déformable.

Cette limitation n'est, en fait, pas très grave. Le but de notre simulateur n'est pas tant de simuler la réalité à tout prix, que de donner à l'utilisateur des sensations proches de celles qu'il rencontrerait dans la réalité. L'évaluation du réalisme d'un simulateur de chirurgie se fait donc essentiellement sur des critères visuels, et haptiques le cas échéant. Nous sommes nous mêmes nos premiers juges, puisque chacun de nos développements est testé pour voir s'il apporte une amélioration significative. La principale source de validation de nos travaux est ensuite le retour que nous avons des autres utilisateurs et en particulier des chirurgiens, dont les impressions et les suggestions sont immédiatement prises en compte afin orienter nos futures recherches.

5.4 Quelques scenarii

La grande flexibilité d'utilisation de ce simulateur est due en grande partie à l'utilisation, pour la partie interface graphique, d'un puissant outil de visualisation développé au sein du projet Epidaure. Ce logiciel ,qui se nomme YAV, est de conception modulaire. Les modules de base permettent la visualisation de scènes 2D, 3D et même 4D. Sur cette base, nous avons développé un module de simulation, qui contient tous les travaux que avons présentés dans ce manuscrit. La grande force de YAV est l'utilisation du langage de *script* Tcl, qui permet d'appeler, par de simples commandes en lignes, les méthodes C++ qui gérent le processus de simulation. La figure 5.5 montre la fenêtre principale de l'interface utilisateur, sur laquelle on peut voir, notamment, la ligne permettant de taper le texte des commandes qui vont contrôler la simulation. On voit, sur la figure 5.6, le caméra qui permet de visualiser la scène dans laquelle sont représentés le modèle déformable et un instrument de chirurgie.

Nous allons maintenant présenter deux applications de notre prototype de simulateur de chirurgie avec retour d'efforts. La première met en œuvre le modèle pré-calculé, dont l'efficacité est mise à profit pour simuler un processus de palpation de la surface d'un organe. La deuxième présente la résection d'une partie du foie à partir du modèle masse tenseur.

5.4.1 Palpation

Le premier scénario de simulation que nous avons choisi met en scène le modèle pré-calculé. Si ce modèle ne permet pas de prendre en compte les découpes, son efficacité, par contre, le rend particulièrement bien adapté pour une application basée sur le retour d'effort. Nous avons donc

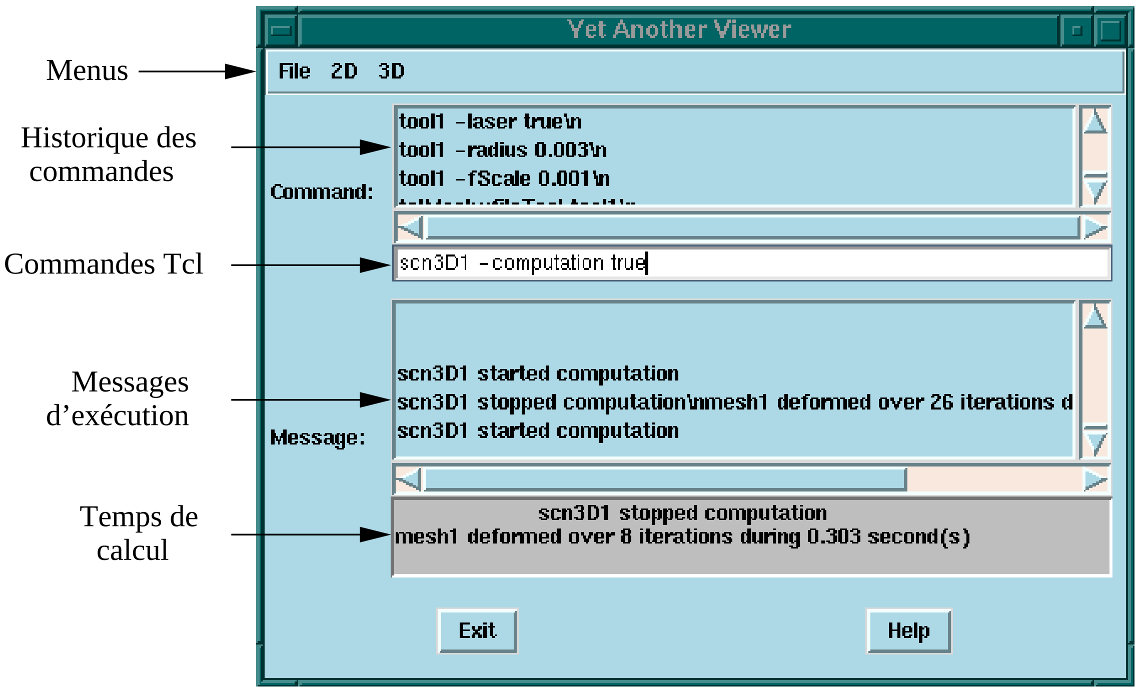


FIG. 5.5 – *Fenêtre principale de l'interface utilisateur.*

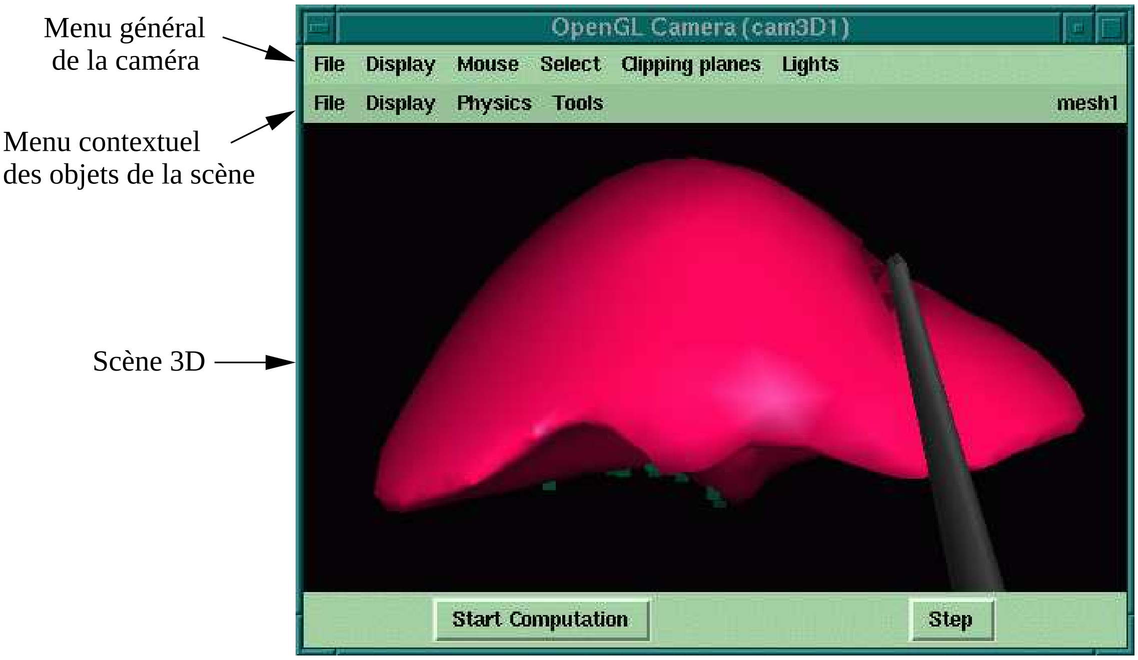


FIG. 5.6 – *Caméra permettant de visualiser la scène 3D.*

mis en place un exercice visant à identifier une zone plus rigide d'un modèle, simplement en faisant glisser un instrument sur sa surface. Ce processus de palpation est souvent utilisé par les médecins, mais aussi par les chirurgiens, pour déceler une anomalie. Dans le cas du foie, cela correspond à la recherche de tumeurs internes dont les tissus sont nettement plus rigides que le parenchyme sain. Nous avons construit des modèles déformables non-homogènes, c'est-à-dire possédant des zones dans lesquelles les coefficients de Lamé sont plus grands. Une telle zone est définie par un ensemble de couches de tétraèdres construites à partir d'un sommet du maillage. On peut voir, sur la figure 5.7, la simulation de tumeurs de trois tailles différentes à partir du même sommet situé à l'intérieur du maillage.

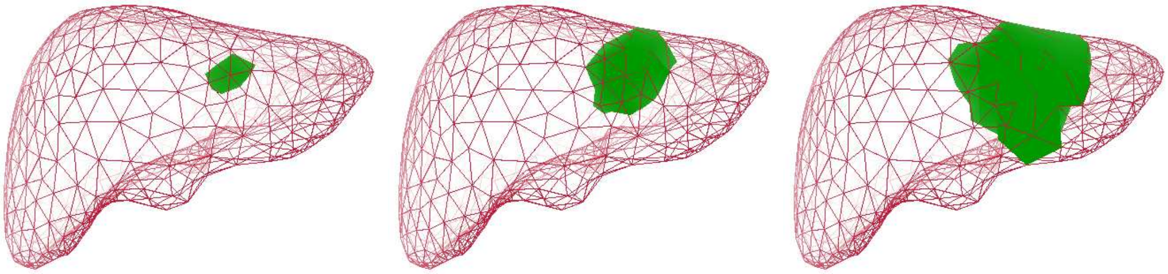


FIG. 5.7 – *Simulation de la présence d'une tumeur : les tétraèdres en verts sont 5 fois plus rigides que le reste du parenchyme.*

Nous avons pré-calculé un modèle de foie possédant la tumeur présentée au milieu de la figure 5.7. La figure 5.8 montre plusieurs étapes du processus de palpation sur ce foie, la ligne bleue représentant la force exercée à chaque instant sur l'instrument. Une certaine dextérité est nécessaire pour réussir à faire glisser l'instrument le long de la surface en appliquant une pression constante. On arrive alors à percevoir assez nettement la rigidité supérieure de la zone représentant la tumeur. Il est aussi possible de détecter la tumeur sans faire glisser l'instrument, simplement en venant exercer des pressions en différents points de la surface.

Comme il n'est pas facile de rendre de manière convaincante la sensation de force ressentie sur le foie, nous proposons un autre exemple. Le modèle est beaucoup plus simple, puisqu'il s'agit d'un parallélépipède, contenant lui aussi une zone plus rigide. On élimine ainsi les variations de forces qui sont dues à la forme complexe de l'objet. On utilise ici un outil virtuel, c'est-à-dire piloté par un fichier. On peut ainsi maîtriser la profondeur d'enfoncement de l'outil dans le modèle, et visualiser (voir figure 5.9) les variations de la force qui est représentée par le trait bleu.

5.4.2 Hépatectomie

La seconde séance de simulation que nous proposons met en œuvre le modèle masse-tenseur. L'exercice consiste ici à découper une partie du foie. L'utilisateur dispose de deux instruments. L'un va venir exercer une traction sur la partie du foie qui doit être supprimée. Nous avons défini deux positions possibles pour l'autre instrument, modélisant ainsi la présence de deux troquarts qui permettent d'accéder à des zones différentes du foie. Ce second outil a le comportement d'un

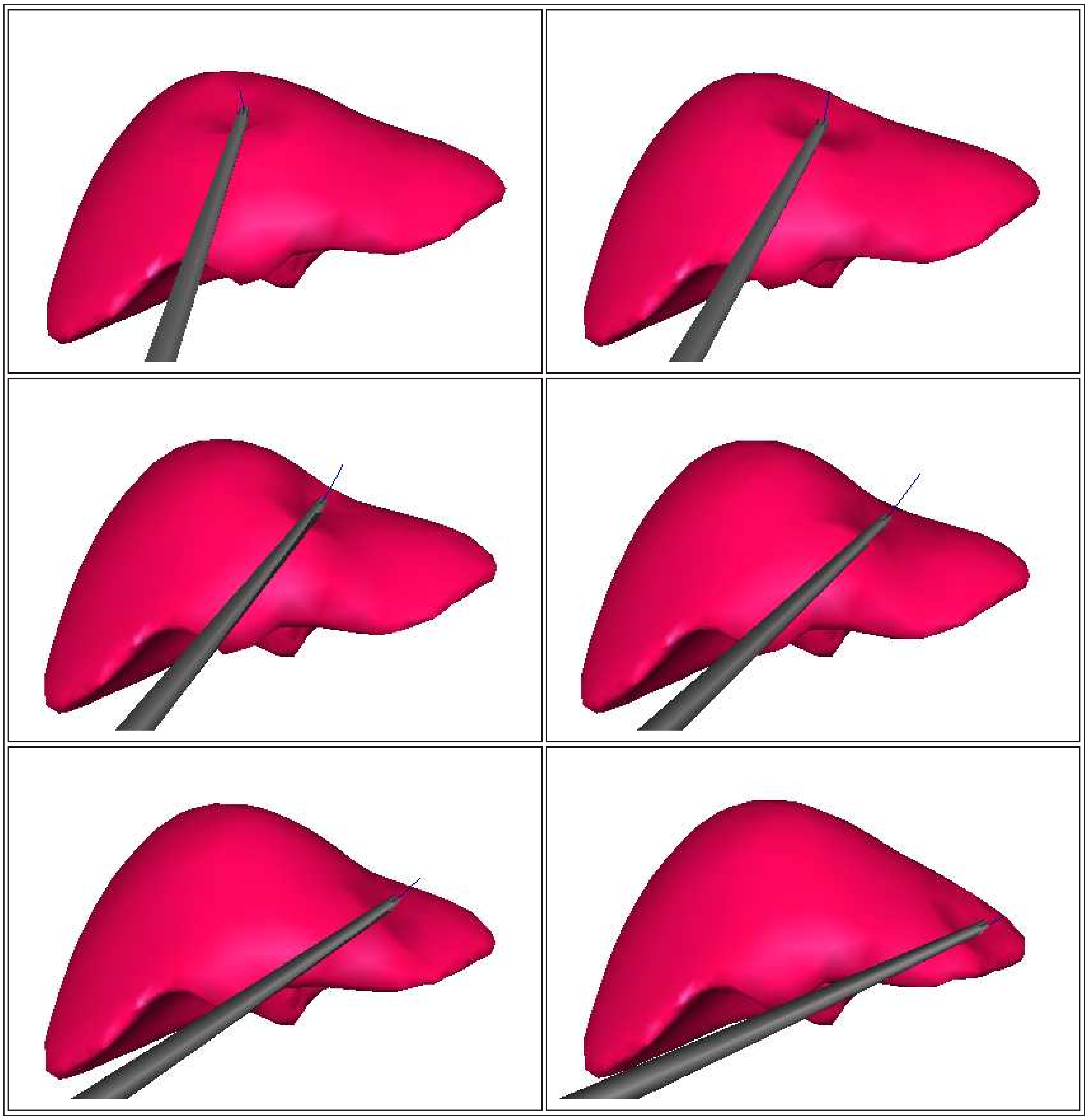


Fig. 5.8 – *Processus de palpation permettant de mettre en évidence la présence d'une tumeur.*

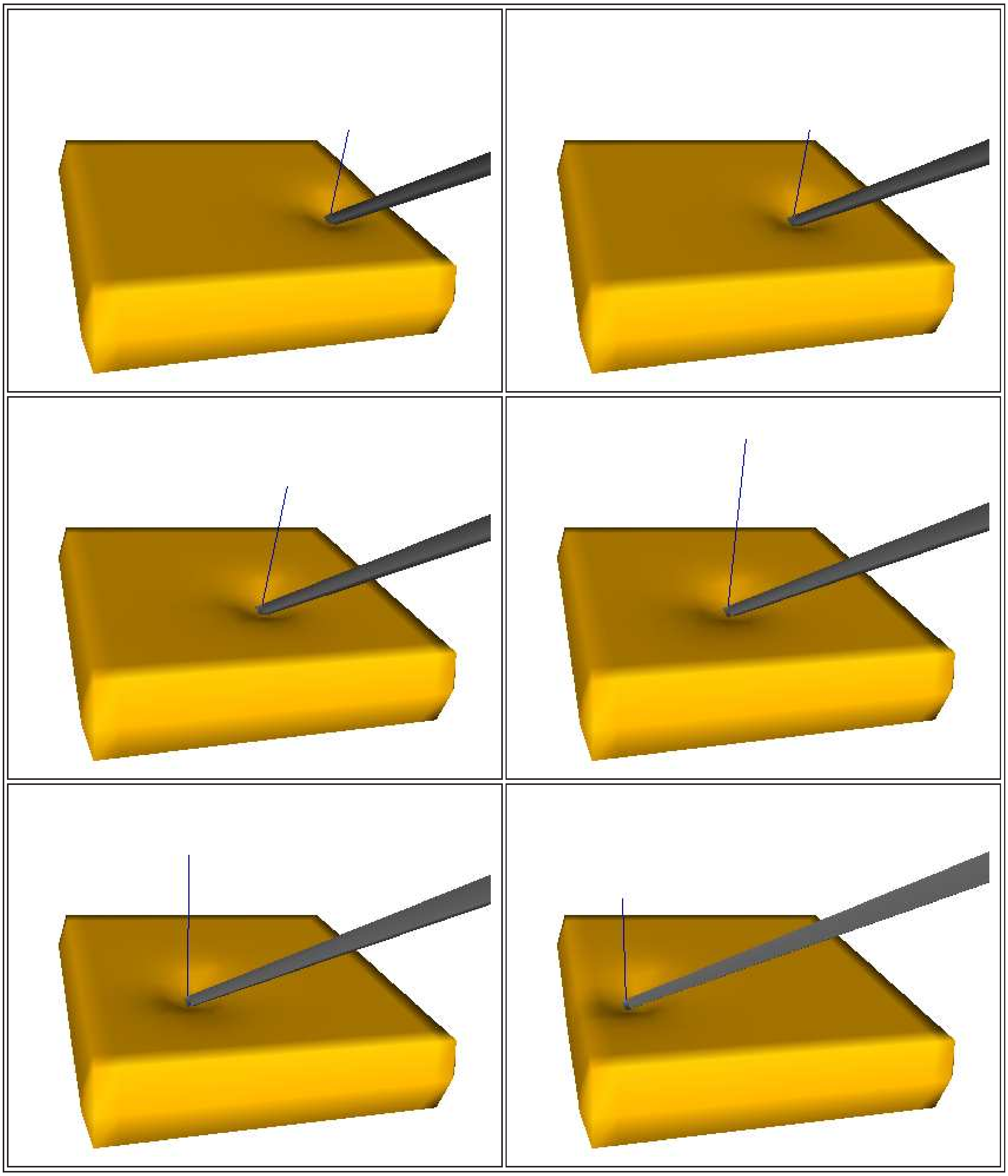


FIG. 5.9 – *Processus de palpation sur un modèle simplifié.*

bistouri à ultrasons, c'est-à-dire qu'il fait exploser toutes les cellules du parenchyme avec lesquelles il entre en contact. Le processus de simulation est alors le suivant:

❶ Le premier instrument vient saisir le bord droit du foie et applique une force de tension qui déforme l'organe.

❷ Une fois que l'organe est sous tension, l'utilisateur commence à pratiquer une incision à l'aide du bistouri. Nous utilisons ici la méthode de raffinement local du maillage qui permet de donner un aspect plus réaliste à la découpe (paragraphe 4.2.4).

❸ Au fur et à mesure que l'incision progresse, les deux parties du foie s'écartent sous l'influence du premier instrument. L'utilisateur a ainsi un meilleur accés pour la suite de la découpe.

❹ Lorsque l'utilisateur juge que la découpe est assez profonde, il change la position de l'instrument. Ceci permet de simuler les changements d'emplacement des outils que pratiquent couramment les chirurgiens lors des opérations.

❺ L'utilisateur peut alors pratiquer une nouvelle incision qui va rejoindre la première afin de totalement séparer le morceau qu'il doit supprimer.

❻ Enfin, la dernière étape consiste à retirer le morceau découpé à l'aide du premier instrument.

5.5 Conclusion

Nous avons détaillé, dans ce dernier chapitre, les différentes étapes nécessaires au réglage d'un simulateur de chirurgie. Nous avons tout d'abord expliqué comment nous reconstruisons les maillages de certains organes à partir d'images médicales. Nous avons ensuite abordé les thèmes de la rhéologie et de l'anatomie, afin de déterminer les caractéristiques qui donnent à nos modèles déformables un aspect réaliste. La second partie du chapitre a été consacrée à la présentation de deux applications, montrant de quelle manière les différents outils que nous avons élaborés durant cette thèse peuvent être assemblés pour simuler certaines procédures chirurgicales.

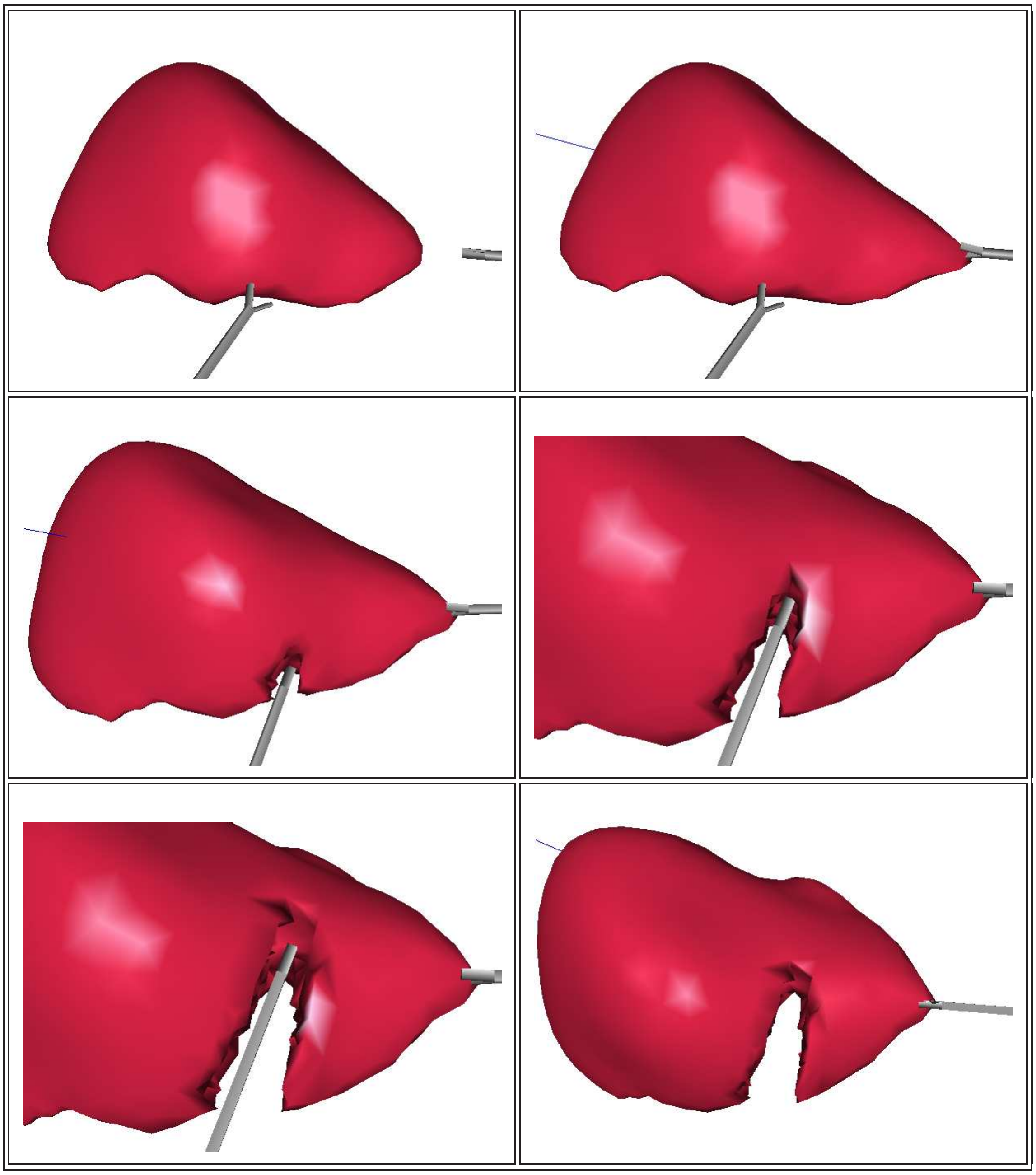


Fig. 5.10 – *Simulation d'une hépatectomie.*

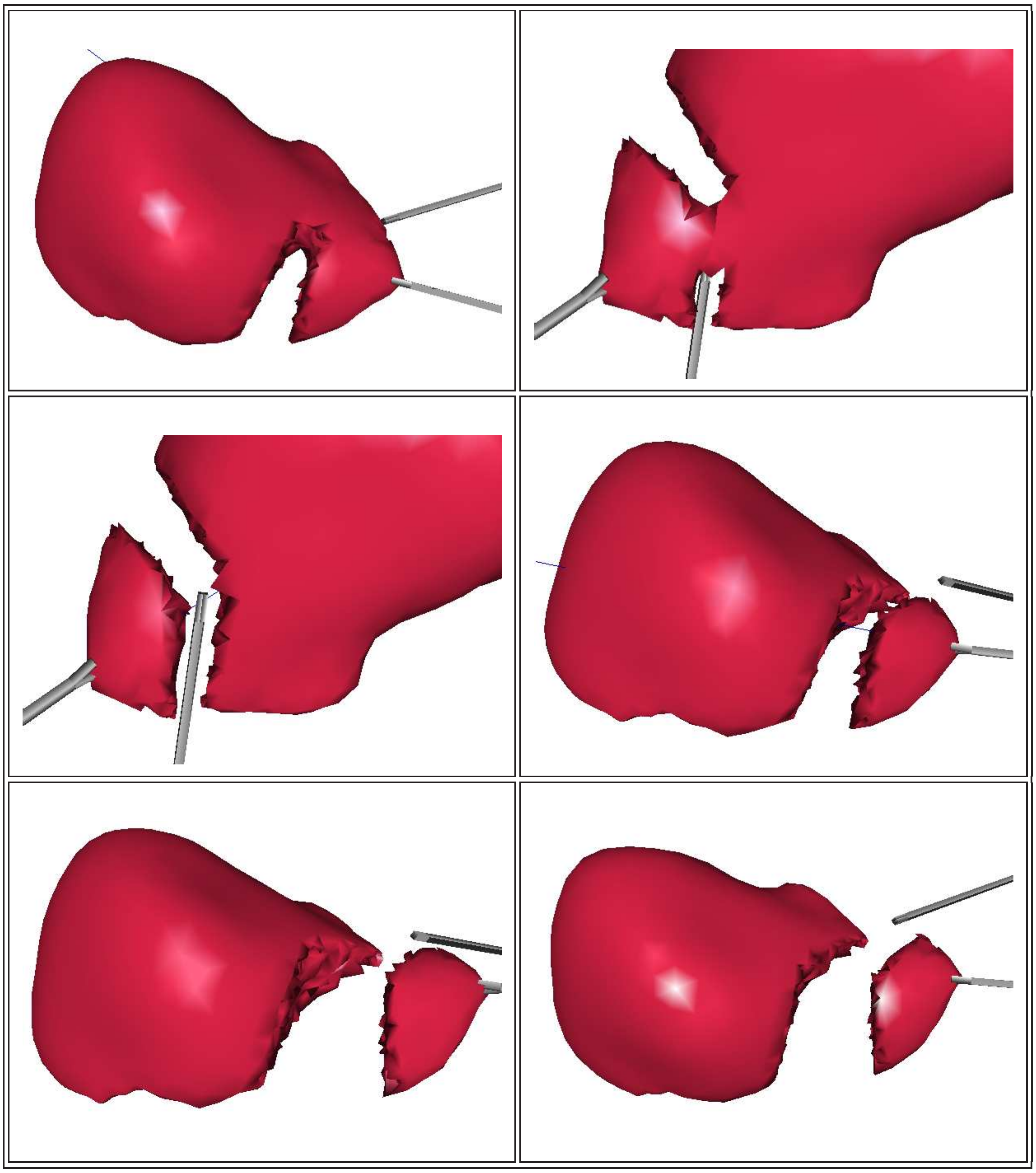


FIG. 5.11 – *Suite de la simulation d'hépatectomie.*

Conclusions et perspectives

Ce dernier chapitre conclut les travaux effectués durant cette thèse. Nous dressons, dans un premier temps, le bilan des recherches réalisées en insistant particulièrement sur les contributions originales que nous avons proposées. Nous détaillons ensuite les points sensibles qui méritent quelques améliorations à court terme. Enfin, nous nous tournerons vers l'avenir en proposant un ensemble de perspectives et de directions de recherche qui nous semblent prometteuses.

« Science is always wrong. It never solves a problem without creating ten more »
George Bernard Shaw (1856-1950),
Irish dramatist and critic.

1 Rappel des contributions

L'élaboration d'un simulateur de chirurgie est un projet de grande ampleur, pour lequel il est nécessaire de combiner des recherches sur plusieurs sujets. On se retrouve, en effet, à la frontière entre les mondes de l'animation, du graphique, du traitement des images, de la biomécanique, du calcul numérique et, bien entendu, de la médecine. La qualité d'un simulateur se jugera à son aptitude à recréer un environnement crédible, qui permettra à l'utilisateur de s'immerger dans l'application qu'on lui propose. La première composante qui conditionnera cette crédibilité sera le réalisme visuel de la scène, ce qui comprend non seulement la qualité de la représentation des objets (organes et instruments de chirurgie), mais aussi l'aspect naturel des mouvements, des déplacements, des déformations et des réactions de ces objets. La seconde composante est le réalisme haptique, c'est-à-dire la qualité de l'information de force qui sera envoyée à l'utilisateur par l'intermédiaire d'une interface spécialisée. Cet aspect prend une dimension particulière lorsque l'on s'adresse à des chirurgiens qui sont particulièrement attentifs aux sensations de toucher. Enfin la dernière composante regroupera la qualité de l'ergonomie générale du simulateur et l'étendue des possibilités offertes. C'est en grande partie ce dernier point qui conditionnera l'intérêt global du simulateur, mais c'est aussi celui qui posera les plus grandes difficultés techniques.

Les travaux que nous avons présentés dans ce manuscrit de thèse portent sur plusieurs aspects qui interviennent dans la construction d'un simulateur de chirurgie. Nous avons ainsi proposé un ensemble d'outils qui permettent, une fois assemblés de construire un prototype de simulateur de chirurgie laparoscopique que nous avons appliqué au foie.

Nous avons consacré une grande partie de nos travaux à l'élaboration et au développement de modèles déformables permettant de représenter les organes du corps humain sur lesquels seront effectués les gestes chirurgicaux. Nous avons tout d'abord présenté en détails deux modèles déformables dérivés des travaux de Stéphane Cotin (Cotin, 1997), qui a initié le sujet de la simulation de chirurgie au sein de notre laboratoire en 1994. Ces modèles sont basés sur la théorie de l'élasticité linéaire et la méthode des éléments finis. Le premier met en œuvre un algorithme de pré-calcul de déformations élémentaires. La linéarité des équations en jeu permet ensuite d'utiliser ces pré-calculs pour exprimer très efficacement n'importe quelle déformation. La contrepartie de cette efficacité est que les pré-calculs ne sont valables que pour une configuration topologique donnée, ce qui interdit de simuler des découpes. C'est pour cette raison que fut développé le modèle masse-tenseur. Il utilise exactement les mêmes équations au sein d'un schéma de résolution explicite dynamique. La résolution du système d'équations linéaires n'étant pas faite à l'avance, ce modèle est moins efficace, mais cela lui permet en revanche, grâce à une structure de données spécialement adaptée, de se prêter de façon naturelle à la simulation des découpes.

Nos efforts se sont alors portés sur l'amélioration du réalisme biomécanique de ces modèles. S'il est relativement aisé de construire un modèle non-homogène, la loi d'élasticité linéaire que nous utilisons se limite aux matériaux isotropes. Les tissus biologiques étant souvent fortement anisotropes, nous avons généralisé nos modèles déformables au cas de l'élasticité transversalement isotrope, qui permet de représenter des matériaux possédant une direction privilégiée de déformation.

Nous avons, de plus, isolé les termes correspondant à l'étirement et au cisaillement provoqués par l'anisotropie. Cela nous permet de choisir le comportement en fonction de la structure que l'on veut modéliser. Cette prise en compte des comportements anisotropes nous a permis de simuler les déformations de structures tubulaires (vaisseaux, tendons), mais aussi de modéliser la présence des gros vaisseaux qui sillonnent le parenchyme hépatique et en modifient le comportement. Afin de se rapprocher encore plus de la réalité anatomique du foie, nous avons voulu tenir compte de la grande rigidité superficielle provoquée par la peau qui entoure l'organe : la capsule de Glisson. Pour cela, nous avons ajouté, à la surface de notre modèle masse-tenseur, des ressorts de fortes raideurs. Nous obtenons ainsi un aspect plus réaliste des déformations, en particulier sur les zones de contact avec les instruments.

Malgré toutes les améliorations que nous leur avons apportées, le réalisme de nos modèles déformables reste limité par l'utilisation de l'élasticité linéaire. En effet, un modèle élastique linéaire n'est pas invariant par rotation, ce qui induit qu'il n'est valable que dans l'hypothèse des petits déplacements. Si les déplacements deviennent trop grands, les rotations que peut subir une partie du modèle engendreront des distortions totalement irréalistes. Pour pallier ce problème, nous avons développé un nouveau modèle déformable de type masse-tenseur non-linéaire, utilisant la loi d'élasticité de St Venant-Kirchhoff, qui est la généralisation de l'élasticité linéaire pour les grands déplacements. Ce modèle est donc invariant par rotation, ce qui améliore grandement le réalisme des déformations dès lors que l'hypothèse des petits déplacements n'est plus vérifiée. Nous nous sommes aussi intéressés au problème de variations de volume en ajoutant à notre modèle des contraintes d'incompressibilité. Afin de bénéficier du même choix dans le comportement des matériaux, nous avons répercuté les améliorations développées sur le modèle linéaire vers ce nouveau modèle. Nous avons, pour cela, étendu l'élasticité de St Venant-Kirchhoff aux matériaux transversalement isotropes, puis nous avons ajouté le modèle masse-ressort surfacique qui représente la capsule de Glisson. La contrepartie de l'amélioration du réalisme étant un surcoût de calcul significatif, nous avons ensuite proposé un modèle dont le comportement s'adapte à l'amplitude des déplacements subis par chaque partie du maillage. Seuls les sommets dont les déplacements sont supérieurs à un seuil utilisent l'élasticité non-linéaire, les autres se contentant d'un comportement linéaire. Nous pouvons donc ainsi bénéficier des qualités du modèle en grands déplacements sans trop pénaliser le fréquence de calcul. Enfin, nous proposons un dernier modèle déformable, issu d'une formulation originale de l'élasticité en grands déplacements au travers de la méthode des éléments finis. Le principal avantage de cette formulation est d'intégrer implicitement l'invariance par rotation, puisque l'énergie élastique ne dépend que de la variation des longueurs des arêtes du maillage. Nous avons développé ce modèle dans le cas des surfaces déformables, puis pour les modèles volumiques.

La seconde partie de nos travaux s'est centrée sur l'interaction entre les modèles déformables et les outils chirurgicaux virtuels manipulés par l'utilisateur. Pour cela, nous avons commencé par développer un modèle de contact. En mettant à profit l'efficacité de l'algorithme de détection de collision que nous utilisons, nous avons défini un schéma de traitement des collisions qui prend en compte la totalité de l'instrument chirurgical et permet d'attribuer des comportements différents au manche et à l'extrémité de l'instrument. À partir de l'interaction de base qui consiste à faire glisser l'outil sur la surface de l'organe, nous avons défini un certain nombre d'actions, comme la possibilité de saisir une zone de la surface à l'aide des mâchoires d'une pince, ou de découper le modèle. Nous proposons d'ailleurs une méthode, basée sur le raffinement local du maillage, pour améliorer le réalisme de la découpe. Nous avons ensuite abordé le problème du couplage du simulateur avec un système à retour d'effort. Ce système mécanique étant relié, pour des raisons matérielles, à un second ordinateur, nous avons dû mettre en place un protocole de communication afin que les boucles asynchrones gérant le modèle déformable d'un coté, et l'instrument à retour d'effort de l'autre, puissent s'échanger les informations de position et de force. Enfin, nous nous sommes intéressés au problème du retour d'effort proprement dit. Si les modèles déformables que nous avons développés sont capables de simuler les déformations et la découpe d'organes en temps-réel visuel (25 Hz), cela ne suffit pas pour obtenir une sensation de retour d'effort réaliste. Le sens du toucher est, en effet, beaucoup plus sensible que celui de la vue, et il faut au moins une fréquence de 300 Hz pour simuler les efforts ressentis lorsque l'on touche un objet mou. Plutôt que de simplifier nos modèles déformables afin qu'il atteignent cette fréquence, nous proposons une méthode d'extrapolation qui permet d'estimer les forces à une fréquence suffisante, à partir des forces calculées par le modèle déformable 25 fois par seconde.

Le dernier chapitre de ce manuscrit propose un petit scénario récapitulant toutes les étapes nécessaires à la mise en œuvre d'une séance de simulation. On peut ainsi se rendre de compte de la façon dont les différentes "briques" que nous avons développées, et que nous venons de récapituler, se réunissent pour former un prototype de simulateur de chirurgie laparoscopique sur le foie, capable modéliser un certain nombre de gestes et de procédures chirurgicales.

2 Perspectives

Si les recherches que nous avons menées nous permettent aujourd'hui de disposer d'un ensemble d'outils performants et bien adaptés à la simulation de chirurgie, il reste encore un travail relativement important d'intégration des différentes composantes. Cette intégration est un travail de tous les instants, qu'il ne faut à aucun moment négliger sous peine de se retrouver dans des impasses techniques, technologiques ou logicielles. Il est de plus nécessaire de se remettre régulièrement en question et de ne pas hésiter à opérer des modifications profondes dans la conception logicielle. Nous sommes d'ailleurs actuellement en train de transférer la majeure partie des travaux que nous avons présentés au sein d'une nouvelle structure de données, mieux adaptée aux développements futurs. Il est également important de porter une attention particulière à l'aspect convivial des logiciels et des interfaces utilisateur. Il est, d'autre part, nécessaire d'adopter des solutions matérielles parfaitement adaptées, pour ne pas réduire à néant tous les efforts de recherche et de développement, mais aussi les plus simples possible. Dans le cas d'un projet comme la simulation de chirurgie, le cahier des charges est assez lourd. Il faut une importante puissance de calcul, la possibilité de faire fonctionner plusieurs processus asynchrones en parallèle, et une puissance graphique suffisante pour ne pas limiter le reste de la simulation. Si on élimine les grosses stations graphiques de type SGI Onyx II, qui sont de prix très élevé, il n'y a aujourd'hui que peu de solutions. Nous venons de recevoir la nouvelle *Visual Workstations* 330[9] développée par SGI, dont le prix est plus raisonnable (dans le cadre d'un équipement hospitalier). Cette machine intègre deux processeurs pentium III et une puissante carte accélératrice 3D développée en collaboration avec la société *nVIDIA*[10]. Nous espérons pouvoir ainsi proposer une solution intégrée sur une seule machine de prix raisonnable. Pour continuer sur l'aspect matériel, nous avons aussi entamé l'intégration d'un système à retour d'effort *Phantom*, que nous comptons utiliser pour de nouvelles applications comme la chirurgie ouverte ou la neurochirurgie.

La seconde partie des améliorations envisageables à court terme concerne plus spécifiquement les modèles déformables que nous avons proposés. Tout d'abord, le modèle de "simplexe hyperélastique", qui a été développé dans la dernière partie de la thèse, demande une étude plus approfondie, afin d'en cerner les caractéristiques et d'en tirer le meilleur. Il sera en particulier intéressant de le généraliser aux matériaux anisotropes. Nous envisageons aussi d'autres techniques de découpe de maillage. Dans le cas de la découpe par retrait de matière nous voulons expérimenter d'autres algorithmes de raffinement local du maillage afin de garantir une meilleure qualité des tétraèdres créés. Afin de s'adapter à d'autres types de chirurgie, il serait aussi intéressant de modéliser les découpes effectuées à l'aide d'un scalpel. En complément de la découpe, un geste qu'il faudra introduire dans le simulateur est celui de la suture. Cela pose deux problèmes principaux. Il faut d'abord être capable d'établir un lien physique entre deux modèles déformables (ou entre deux parties du même modèle). Se pose ensuite le problème de la modélisation du fil ou des agrafes.

Enfin, une dernière composante, qui s'inscrit peut être à un peu plus long terme, sera de définir un ensemble de possibilités de couplage entre plusieurs modèles déformables. Si aujourd'hui rien ne nous empêche de placer plusieurs modèles dans la même scène et d'agir avec chacun d'eux avec les mêmes outils, ces modèles se comporteront comme s'ils se trouvaient seuls. On peut envisager alors deux modes d'interaction possibles. Le premier cas, qui est le plus simple, serait de définir une relation topologique entre deux modèles. Il faut alors simplement être capable d'échanger entre les deux maillages des informations de force et de position. On pourrait ainsi imaginer attacher le foie à un modèle des vaisseaux qui y arrivent et en repartent. Cette solution serait, anatomiquement parlant, beaucoup plus réaliste que de fixer le foie dans l'espace. Enfin, la deuxième amélioration intéressante serait de modéliser la collision et le contact entre deux modèles déformables, ce qui est un projet beaucoup plus ambitieux.

Si l'on considère maintenant de manière plus globale quels sont les développements qui sont nécessaires à l'amélioration du réalisme de notre simulateur, on peut placer en haut de la liste la définition d'un environnement anatomique. Ce point comprend l'intégration des organes et des structures anatomiques qui entourent la zone d'intérêt. Il est également important travailler le réalisme visuel de la scène à l'aide de textures. L'étape suivante consiste à introduire une composante

9. SGI 330 : http://www.sgi.com/workstations/330/index.html
10. nVIDIA : http://www.nvidia.com/

physiologique à la simulation. Des effets spéciaux permettront, par exemple, de simuler l'écoulement du sang, ou la fumée qui accompagne la coagulation. Il faudra aussi tenir compte de la respiration du patient, qui fait se déplacer un certain nombre d'organes, ou encore les battements cardiaques qui se propagent dans les artères. Tous ces aspects supplémentaires augmenteront énormément la puissance de calcul nécessaire, ce qui amènera immanquablement à envisager la parallélisation des logiciels.

Si l'on reste dans le cadre de l'amélioration du réalisme de la simulation, on ne peut pas ignorer la prise en compte de l'environnement opératoire. Sans aller jusqu'à recréer une salle d'opération, il faut apporter un soin tout particulier à l'ergonomie, afin de replonger l'utilisateur dans des conditions proche de la réalité.

Un bon simulateur de chirurgie doit permettre à l'utilisateur de progresser et d'améliorer sa technique. Il est donc nécessaire pour cela d'évaluer les prestations, sur la base de critères définis à l'avance. Après chaque entraînement, le simulateur pourra alors fournir à l'utilisateur un bilan des points forts et des points faibles des gestes qu'il vient d'exécuter. À l'inverse, il est aussi possible de demander au simulateur de surveiller un point particulier, ce qui permettrait par exemple de vérifier qu'un nouveau geste ne va pas entraîner des contraintes trop importantes sur une zone bien précise d'un organe.

Enfin, la dernière étape dans l'évolution d'un simulateur de chirurgie sera son évaluation en milieu hospitalier. Il sera alors important de bien définir le protocole d'évaluation, afin de cerner le mieux possible les points critiques qui limitent la qualité globale, et donc l'intérêt pratique, du simulateur.

Annexes

Annexe A

Quelques calculs théoriques

A.1 Formulation variationnelle du problème élastique linéaire

Le calcul de la première variation de l'énergie élastique de déformation, dans le cas linéaire, se fait de la manière suivante :

$$\delta W_l = div\left[\left(\frac{\partial W^l}{\partial \nabla \mathbf{U}}\right)^t\right] \qquad \text{avec} \qquad W^l = \frac{\lambda}{2}(div\ \mathbf{U})^2 + \mu\|\nabla \mathbf{U}\|^2 - \frac{\mu}{2}\|rot\ \mathbf{U}\|^2 \tag{A.1}$$

On va donc appliquer ce calcul successivement aux différents termes de l'énergie :

- Premier terme :

$$\frac{\partial div \mathbf{U}}{\partial \nabla \mathbf{U}} = \frac{\partial(u_x + v_y + w_z)}{\partial \begin{bmatrix} u_x & u_y & u_z \\ v_x & v_y & v_z \\ w_x & w_y & w_z \end{bmatrix}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{A.2}$$

$$div(kM) = k\ div\ M + M\ \nabla k, \tag{A.3}$$

$$\text{donc} \qquad \boxed{\frac{\lambda}{2}(div\ \mathbf{U})^2 \xrightarrow{\left(\frac{\partial}{\partial \nabla \mathbf{U}}\right)^t} \lambda\ div \mathbf{U}\ I \xrightarrow{div} \lambda\ \nabla\left(div \mathbf{U}\right)}. \tag{A.4}$$

- Second terme :

$$\boxed{\mu\|\nabla \mathbf{U}\|^2 \xrightarrow{\left(\frac{\partial}{\partial \nabla \mathbf{U}}\right)^t} 2\mu\ \nabla \mathbf{U}^t \xrightarrow{div} 2\mu\ \Delta \mathbf{U}}. \tag{A.5}$$

- Troisième terme :

$$
\begin{aligned}
\frac{\partial \|rot\mathbf{U}\|^2}{\partial \nabla \mathbf{U}} &= \frac{\partial\left(u_y^2+u_z^2+v_x^2+v_z^2+w_x^2+w_y^2-2u_yv_x-2u_zw_x-2v_zw_y\right)}{\partial \begin{bmatrix} u_x & u_y & u_z \\ v_x & v_y & v_z \\ w_x & w_y & w_z \end{bmatrix}} \\
&= \begin{bmatrix} 0 & 2(u_y-v_x) & 2(u_z-w_x) \\ 2(v_x-u_y) & 0 & 2(v_z-w_y) \\ 2(w_x-u_z) & 2(w_y-v_z) & 0 \end{bmatrix} = 2\left(\nabla\mathbf{U}-\nabla\mathbf{U}^t\right) \qquad \text{(A.6)}
\end{aligned}
$$

$$
\text{donc} \quad \boxed{-\mu\|rot\mathbf{U}\|^2 \xrightarrow{\left(\frac{\partial}{\partial\nabla\mathbf{U}}\right)^t} -\mu\ \left(\nabla\mathbf{U}^t+\nabla\mathbf{U}\right) \xrightarrow{div} -\mu\ \Delta\mathbf{U}+\mu\ \nabla\left(div\mathbf{U}\right)}. \qquad \text{(A.7)}
$$

A.2 L'énergie de déformation de St Venant-Kirchhoff

Dans ce paragraphe, nous détaillons les calculs permettant d'exprimer l'énergie de déformation de St Venant-Kirchhoff en fonction du gradient du champ de déplacement $\nabla\mathbf{U}$. Pour cela il est plus simple de commencer par exprimer l'énergie de déformation en fonction du tenseur de déformation de Cauchy-Green C, pour ensuite revenir à E.

Nous rappelons tout d'abord que :

$$
E = \frac{1}{2}\big(C-I\big) = \frac{1}{2}\big(\nabla\mathbf{\Phi}^t\nabla\mathbf{\Phi}-I\big), \qquad \text{(A.8)}
$$

où $\nabla\mathbf{\Phi}$ est le gradient de la déformée. Alors l'énergie de déformation W s'exprime par :

$$
\begin{aligned}
W &= \frac{\lambda}{2}\left(trE\right)^2+\mu\ trE^2 \\
&= \frac{\lambda}{8}\left(tr(C-I)\right)^2+\frac{\mu}{4}\ tr\big(C-I\big)^2 \\
&= \frac{\lambda}{8}\left(tr(\nabla\mathbf{\Phi}^t\nabla\mathbf{\Phi})-3\right)^2+\frac{\mu}{4}\ tr\big(\nabla\mathbf{\Phi}^t\nabla\mathbf{\Phi}\nabla\mathbf{\Phi}^t\nabla\mathbf{\Phi}-2\nabla\mathbf{\Phi}^t\nabla\mathbf{\Phi}+I\big) \\
&= \frac{\lambda}{8}\left(\|\nabla\mathbf{\Phi}\|^2-3\right)^2+\frac{\mu}{4}\left(\|\nabla\mathbf{\Phi}^t\nabla\mathbf{\Phi}\|^2-2\|\nabla\mathbf{\Phi}\|^2+3\right) \qquad \text{(A.9)}
\end{aligned}
$$

On utilise ensuite les équations suivantes pour passer de $\nabla\mathbf{\Phi}$ à $\nabla\mathbf{U}$:

$$\begin{aligned}
\|\nabla\mathbf{\Phi}\|^2 &= \|\nabla\mathbf{U} + I\|^2 \\
&= \|\nabla\mathbf{U}\|^2 + 2div\mathbf{U} + 3 \qquad \text{(A.10)}\\
\|\nabla\mathbf{\Phi}^t\nabla\mathbf{\Phi}\|^2 &= \|\nabla\mathbf{U}^t\nabla\mathbf{U} + \nabla\mathbf{U}^t + \nabla\mathbf{U} - I\|^2 \\
&= \|\nabla\mathbf{U}^t\nabla\mathbf{U}\|^2 + 2\|\nabla\mathbf{U}\|^2 + \|I\|^2 + \underbrace{2(\nabla\mathbf{U}^t\nabla\mathbf{U} : \nabla\mathbf{U}^t) + 2(\nabla\mathbf{U}^t\nabla\mathbf{U} : \nabla\mathbf{U})}_{4(\nabla\mathbf{U} : \nabla\mathbf{U}^t\nabla\mathbf{U})} \\
&+ \underbrace{2(\nabla\mathbf{U}^t\nabla\mathbf{U} : I)}_{2\|\nabla\mathbf{U}\|^2} + \underbrace{2(\nabla\mathbf{U}^t : \nabla\mathbf{U})}_{\substack{2\|\nabla\mathbf{U}\|^2 \\ -2\|rot\ \mathbf{U}\|^2}} + \underbrace{2(\nabla\mathbf{U}^t : I) + 2(\nabla\mathbf{U} : I)}_{4div\ \mathbf{U}} \\
&= \|\nabla\mathbf{U}^t\nabla\mathbf{U}\|^2 + 4(\nabla\mathbf{U} : \nabla\mathbf{U}^t\nabla\mathbf{U}) + 6\|\nabla\mathbf{U}\|^2 - 2\|rot\ \mathbf{U}\|^2 + 4div\ \mathbf{U} + 3 \qquad \text{(A.11)}
\end{aligned}$$

Ce qui permet d'exprimer les deux termes de l'équation A.9 sous la forme :

$$\begin{aligned}
\frac{\lambda}{8}\left(\|\nabla\mathbf{\Phi}\|^2 - 3\right)^2 &= \frac{\lambda}{8}\left(2div\mathbf{U} + \|\nabla\mathbf{U}\|^2\right)^2 \\
&= \frac{\lambda}{2}\left(div\mathbf{U} + \frac{1}{2}\|\nabla\mathbf{U}\|^2\right)^2 \qquad \text{(A.12)}\\
\frac{\mu}{4}\left(\|\nabla\mathbf{\Phi}^t\nabla\mathbf{\Phi}\|^2 - 2\|\nabla\mathbf{\Phi}\|^2 + 3\right) &= \mu\|\nabla\mathbf{U}\|^2 - \frac{\mu}{2}\|rot\ \mathbf{U}\|^2 \\
&+ \mu(\nabla\mathbf{U} : \nabla\mathbf{U}^t\nabla\mathbf{U}) + \frac{\mu}{4}\|\nabla\mathbf{U}^t\nabla\mathbf{U}\|^2 \qquad \text{(A.13)}
\end{aligned}$$

Annexe B

Calculs sur les éléments finis

B.1 L'élasticité linéaire transversalement isotrope

L'application de la méthode des éléments finis à l'élasticité linéaire transversalement isotrope impose d'exprimer les nouveaux termes I_4^2, $I_1 I_4$ et I_5 en fonction de la direction privilégiée $\mathbf{a}_0$ et des vecteurs de forme $\{\boldsymbol{\alpha}_j, j = 0..3\}$ du tétraèdre. On rappelle tout d'abord les expressions du tenseur de déformation, E_l, et de sa trace :

$$E_l = \frac{1}{2} \sum_j \left(\mathbf{U}_j \otimes \boldsymbol{\alpha}_j + \boldsymbol{\alpha}_j \otimes \mathbf{U}_j \right) \quad \text{et} \quad tr E_l = \sum_j \mathbf{U}_j \,.\, \boldsymbol{\alpha}_j \tag{B.1}$$

On peut alors calculer les termes anisotropes :

$$\begin{aligned}
I_4^2 &= \left(\mathbf{a}_0^t \; E_l \; \mathbf{a}_0 \right)^2 = \left(\sum_j \mathbf{a}_0^t \; \mathbf{U}_j \; \boldsymbol{\alpha}_j^t \; \mathbf{a}_0 \right)^2 \\
&= \sum_{j,k} \left(\mathbf{a}_0^t \; \mathbf{U}_j \right) \left(\boldsymbol{\alpha}_j^t \; \mathbf{a}_0 \right) \left(\mathbf{a}_0^t \; \mathbf{U}_k \right) \left(\boldsymbol{\alpha}_k^t \; \mathbf{a}_0 \right) \\
&= \sum_{j,k} \left(\mathbf{U}_j^t \; \mathbf{a}_0 \right) \left(\mathbf{a}_0^t \; \boldsymbol{\alpha}_j \right) \left(\boldsymbol{\alpha}_k^t \; \mathbf{a}_0 \right) \left(\mathbf{a}_0^t \; \mathbf{U}_k \right) \\
I_4^2 &= \sum_{j,k} \mathbf{U}_j^t \left[\left(\mathbf{a}_0 \otimes \mathbf{a}_0 \right) \left(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k \right) \left(\mathbf{a}_0 \otimes \mathbf{a}_0 \right) \right] \mathbf{U}_k
\end{aligned} \tag{B.2}$$

$$\begin{aligned}
I_1 I_4 &= tr E_l \; \left(\mathbf{a}_0^t \; E_l \; \mathbf{a}_0 \right) = \left(\sum_j \mathbf{U}_j \,.\, \boldsymbol{\alpha}_j \right) \left(\sum_k \mathbf{a}_0^t \; \mathbf{U}_k \; \boldsymbol{\alpha}_k^t \; \mathbf{a}_0 \right) \\
&= \sum_{j,k} \mathbf{U}_j^t \; \boldsymbol{\alpha}_j \; \boldsymbol{\alpha}_k^t \; \mathbf{a}_0 \; \mathbf{a}_0^t \; \mathbf{U}_k \\
I_1 I_4 &= \sum_{j,k} \mathbf{U}_j^t \left[\left(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k \right) \left(\mathbf{a}_0 \otimes \mathbf{a}_0 \right) \right] \mathbf{U}_k
\end{aligned} \tag{B.3}$$

$$
\begin{aligned}
I_5 &= \mathbf{a}_0^t\ E_l^2\ \mathbf{a}_0 \\
&= \mathbf{a}_0^t \left(\frac{1}{4} \sum_{j,k} \mathbf{U}_j\ \boldsymbol{\alpha}_j^t\ \mathbf{U}_k\ \boldsymbol{\alpha}_k^t + \boldsymbol{\alpha}_j\ \mathbf{U}_j^t\ \boldsymbol{\alpha}_k\ \mathbf{U}_k^t + \mathbf{U}_j\ \boldsymbol{\alpha}_j^t\ \boldsymbol{\alpha}_k\ \mathbf{U}_k^t + \boldsymbol{\alpha}_j\ \mathbf{U}_j^t\ \mathbf{U}_k\ \boldsymbol{\alpha}_k^t \right) \mathbf{a}_0 \\
&= \frac{1}{4} \sum_{j,k} \big(\mathbf{a}_0^t\ \mathbf{U}_j\big)\ \big(\boldsymbol{\alpha}_j^t\ \mathbf{U}_k\big)\ \big(\boldsymbol{\alpha}_k^t\ \mathbf{a}_0\big) + \big(\mathbf{a}_0^t\ \boldsymbol{\alpha}_j\big)\ \big(\mathbf{U}_j^t\ \boldsymbol{\alpha}_k\big)\ \big(\mathbf{U}_k^t\ \mathbf{a}_0\big) \\
&\qquad + \big(\mathbf{a}_0^t\ \mathbf{U}_j\big)\ \big(\boldsymbol{\alpha}_j^t\ \boldsymbol{\alpha}_k\big)\ \big(\mathbf{U}_k^t\ \mathbf{a}_0\big) + \big(\mathbf{a}_0^t\ \boldsymbol{\alpha}_j\big)\ \big(\mathbf{U}_j^t\ \mathbf{U}_k\big)\ \big(\boldsymbol{\alpha}_k^t\ \mathbf{a}_0\big) \\
&= \frac{1}{4} \sum_{j,k} \big(\mathbf{U}_j^t\ \mathbf{a}_0\big)\ \big(\mathbf{a}_0^t\ \boldsymbol{\alpha}_k\big)\ \big(\boldsymbol{\alpha}_j^t\ \mathbf{U}_k\big) + \big(\mathbf{U}_j^t\ \boldsymbol{\alpha}_k\big)\ \big(\boldsymbol{\alpha}_j^t\ \mathbf{a}_0\big)\ \big(\mathbf{a}_0^t\ \mathbf{U}_k\big) \\
&\qquad + \big(\mathbf{U}_j^t\ \mathbf{a}_0\big)\ \big(\boldsymbol{\alpha}_j^t\ \boldsymbol{\alpha}_k\big)\ \big(\mathbf{a}_0^t\ \mathbf{U}_k\big) + \mathbf{U}_j^t\ \big(\mathbf{a}_0^t\ \boldsymbol{\alpha}_j\big)\ \big(\mathbf{a}_0^t\ \boldsymbol{\alpha}_k\big)\ \mathbf{U}_k \\
I_5 &= \frac{1}{4} \sum_{j,k} \mathbf{U}_j^t\ \Big[\big(\mathbf{a}_0 \otimes \mathbf{a}_0\big)\ \big(\boldsymbol{\alpha}_k \otimes \boldsymbol{\alpha}_j\big) + \big(\boldsymbol{\alpha}_k \otimes \boldsymbol{\alpha}_j\big)\ \big(\mathbf{a}_0 \otimes \mathbf{a}_0\big) \\
&\qquad\qquad + \big(\boldsymbol{\alpha}_j\ .\ \boldsymbol{\alpha}_k\big)\ \big(\mathbf{a}_0 \otimes \mathbf{a}_0\big) + \big(\mathbf{a}_0 \otimes \mathbf{a}_0\big) : \big(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k\big) I \Big]\ \mathbf{U}_k
\end{aligned}
\tag{B.4}
$$

B.2 L'élasticité non-linéaire de St Venant-Kirchhoff

On rappelle tout d'abord que le gradient du champ de déplacement s'exprime, en fonction des vecteurs de forme du tétraèdre, de la manière suivante :

$$\nabla \mathbf{U} = \sum_j \mathbf{U}_j \otimes \alpha_j$$

Ce qui permet d'obtenir l'expression du tenseur de déformation de Green-St Venant E :

$$E = \frac{1}{2} \left[\nabla \mathbf{U}^t + \nabla \mathbf{U} + \nabla \mathbf{U}^t \nabla \mathbf{U} \right] \tag{B.6}$$

$$= \frac{1}{2} \sum_j \left(\mathbf{U}_j \otimes \alpha_j + \alpha_j \otimes \mathbf{U}_j \right) + \frac{1}{2} \sum_j \alpha_j \otimes \mathbf{U}_j \sum_k \mathbf{U}_k \otimes \alpha_k \tag{B.7}$$

$$= \frac{1}{2} \sum_j \left(\mathbf{U}_j \otimes \alpha_j + \alpha_j \otimes \mathbf{U}_j \right) + \frac{1}{2} \sum_{j,k} (\mathbf{U}_j . \mathbf{U}_k)(\alpha_j \otimes \alpha_k) \tag{B.8}$$

L'énergie élastique de déformation W s'écrit :

$$W = \frac{\lambda}{2} \left(trE \right)^2 + \mu tr E^2 \tag{B.9}$$

Calculons d'abord trE :

$$E = \frac{1}{2} \sum_j \left[tr \left(\mathbf{U}_j \otimes \boldsymbol{\alpha}_j \right) + tr \left(\boldsymbol{\alpha}_j \otimes \mathbf{U}_j \right) \right] + \frac{1}{2} \sum_{j,k} tr \left[(\mathbf{U}_j . \mathbf{U}_k)(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k) \right] \tag{B.10}$$

Or,

$$\left\{ \begin{array}{l} tr \left(\mathbf{U}_j \otimes \boldsymbol{\alpha}_j \right) = \mathbf{U}_j . \boldsymbol{\alpha}_j \\ \\ tr \left[(\mathbf{U}_j . \mathbf{U}_k)(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k) \right] = (\mathbf{U}_j . \mathbf{U}_k) tr (\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k) = (\mathbf{U}_j . \mathbf{U}_k)(\boldsymbol{\alpha}_j . \boldsymbol{\alpha}_k) \end{array} \right. \tag{B.11}$$

Donc,

$$trE = \sum_j \mathbf{U}_j.\boldsymbol{\alpha}_j + \frac{1}{2}\sum_{j,k}(\mathbf{U}_j.\mathbf{U}_k)(\boldsymbol{\alpha}_j.\boldsymbol{\alpha}_k) \tag{B.12}$$

On en déduit $(trE)^2$:

$$(trE)^2 = \underbrace{\left(\sum_j \mathbf{U}_j.\boldsymbol{\alpha}_j\right)^2}_{(1)} + \underbrace{\left(\sum_j \mathbf{U}_j.\boldsymbol{\alpha}_j\right)\left[\sum_{j,k}(\mathbf{U}_j.\mathbf{U}_k)(\boldsymbol{\alpha}_j.\boldsymbol{\alpha}_k)\right]}_{(2)} + \underbrace{\frac{1}{4}\left[\sum_{j,k}(\mathbf{U}_j.\mathbf{U}_k)(\boldsymbol{\alpha}_j.\boldsymbol{\alpha}_k)\right]^2}_{(3)} \tag{B.13}$$

Avec les différents termes qui valent :

$$(1) = \sum_{j,k} \mathbf{U}_j^t\boldsymbol{\alpha}_j\mathbf{U}_k^t\boldsymbol{\alpha}_k = \sum_{j,k} \mathbf{U}_j^t\left(\boldsymbol{\alpha}_j\boldsymbol{\alpha}_k^t\right)\mathbf{U}_k = \sum_{j,k} \mathbf{U}_j^t\left(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k\right)\mathbf{U}_k \tag{B.14}$$

$$(2) = \sum_{j,k,l} (\mathbf{U}_j.\boldsymbol{\alpha}_j)(\mathbf{U}_k.\mathbf{U}_l)(\boldsymbol{\alpha}_k.\boldsymbol{\alpha}_l) \tag{B.15}$$

$$(3) = \frac{1}{4}\sum_{j,k,l,m} (\mathbf{U}_j.\mathbf{U}_k)(\boldsymbol{\alpha}_j.\boldsymbol{\alpha}_k)(\mathbf{U}_l.\mathbf{U}_m)(\boldsymbol{\alpha}_l.\boldsymbol{\alpha}_m) \tag{B.16}$$

Calculons ensuite E^2 :

$$E^2 = \underbrace{\frac{1}{4}\left[\sum_j (\mathbf{U}_j \otimes \boldsymbol{\alpha}_j + \boldsymbol{\alpha}_j \otimes \mathbf{U}_j)\right]^2}_{(1)} + \underbrace{\frac{1}{4}\left[\sum_{j,k}(\mathbf{U}_j.\mathbf{U}_k)(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k)\right]^2}_{(3)} \tag{B.17}$$

$$+ \underbrace{\frac{1}{2}\left[\sum_j (\mathbf{U}_j \otimes \boldsymbol{\alpha}_j + \boldsymbol{\alpha}_j \otimes \mathbf{U}_j)\right]\left[\sum_{k,l}(\mathbf{U}_k.\mathbf{U}_l)(\boldsymbol{\alpha}_k \otimes \boldsymbol{\alpha}_l)\right]}_{(2)} \tag{B.18}$$

Où chacun des termes vaut :

$$(1) = \frac{1}{4}\sum_{j,k} 2\left(\mathbf{U}_j \otimes \boldsymbol{\alpha}_k\right)\left(\mathbf{U}_k.\boldsymbol{\alpha}_j\right) + \left(\mathbf{U}_j \otimes \mathbf{U}_k\right)\left(\boldsymbol{\alpha}_j.\boldsymbol{\alpha}_k\right) + \left(\mathbf{U}_j.\mathbf{U}_k\right)\left(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k\right) \quad \text{(B.19)}$$

$$tr(1) = \frac{1}{4}\sum_{j,k} 2\left(\mathbf{U}_j.\boldsymbol{\alpha}_k\right)\left(\mathbf{U}_k.\boldsymbol{\alpha}_j\right) + \left(\mathbf{U}_j.\mathbf{U}_k\right)\left(\boldsymbol{\alpha}_j.\boldsymbol{\alpha}_k\right) + \left(\mathbf{U}_j.\mathbf{U}_k\right)\left(\boldsymbol{\alpha}_j.\boldsymbol{\alpha}_k\right) \quad \text{(B.20)}$$

$$= \frac{1}{2}\sum_{j,k} \mathbf{U}_j^t\left(\boldsymbol{\alpha}_k \otimes \boldsymbol{\alpha}_j\right)\mathbf{U}_k + \mathbf{U}_j^t\left[\left(\boldsymbol{\alpha}_j.\boldsymbol{\alpha}_k\right)Id_3\right]\mathbf{U}_k \quad \text{(B.21)}$$

$$(2) = \frac{1}{2}\sum_{j,k,l} \left(\mathbf{U}_k.\mathbf{U}_l\right)\left(\mathbf{U}_j \otimes \boldsymbol{\alpha}_j\right)\left(\boldsymbol{\alpha}_k \otimes \boldsymbol{\alpha}_l\right) + \left(\mathbf{U}_k.\mathbf{U}_l\right)\left(\boldsymbol{\alpha}_j \otimes \mathbf{U}_j\right)\left(\boldsymbol{\alpha}_k \otimes \boldsymbol{\alpha}_l\right) \quad \text{(B.22)}$$

$$tr(2) = \frac{1}{2}\sum_{j,k,l} \left(\mathbf{U}_k.\mathbf{U}_l\right)\left(\mathbf{U}_j.\boldsymbol{\alpha}_l\right)\left(\boldsymbol{\alpha}_j.\boldsymbol{\alpha}_k\right) + \left(\mathbf{U}_k.\mathbf{U}_l\right)\left(\mathbf{U}_j.\boldsymbol{\alpha}_k\right)\left(\boldsymbol{\alpha}_j.\boldsymbol{\alpha}_l\right) \quad \text{(B.23)}$$

$$(3) = \frac{1}{4}\sum_{j,k,l,m} (\mathbf{U}_j.\mathbf{U}_k)(\mathbf{U}_l.\mathbf{U}_m)(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k)(\boldsymbol{\alpha}_l \otimes \boldsymbol{\alpha}_m) \quad \text{(B.24)}$$

$$tr(3) = \frac{1}{4}\sum_{j,k,l,m} (\mathbf{U}_j.\mathbf{U}_k)(\mathbf{U}_l.\mathbf{U}_m)(\boldsymbol{\alpha}_j.\boldsymbol{\alpha}_m)(\boldsymbol{\alpha}_k.\boldsymbol{\alpha}_l) \quad \text{(B.25)}$$

On a donc, finalement, l'équation globale de l'énergie potentielle élastique :

$$\begin{aligned}
W &= \frac{\lambda}{2}(trE)^2 + \mu trE^2 && \text{(B.26)}\\
&= \frac{\lambda}{2}\sum_{j,k}\left[\mathbf{U}_j^t\left(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k\right)\mathbf{U}_k\right] + \frac{\mu}{2}\sum_{j,k}\left[\mathbf{U}_j^t\left(\boldsymbol{\alpha}_k \otimes \boldsymbol{\alpha}_j\right)\mathbf{U}_k + \mathbf{U}_j^t\left[\left(\boldsymbol{\alpha}_j.\boldsymbol{\alpha}_k\right)Id_3\right]\mathbf{U}_k\right] && \text{(B.27)}\\
&+ \frac{\lambda}{2}\sum_{j,k,l}\left(\mathbf{U}_k.\mathbf{U}_l\right)\left(\mathbf{U}_j.\boldsymbol{\alpha}_j\right)\left(\boldsymbol{\alpha}_k.\boldsymbol{\alpha}_l\right) + \frac{\mu}{2}\sum_{j,k,l}\left(\mathbf{U}_k.\mathbf{U}_l\right)\left(\mathbf{U}_j.\boldsymbol{\alpha}_l\right)\left(\boldsymbol{\alpha}_j.\boldsymbol{\alpha}_k\right) + \left(\mathbf{U}_k.\mathbf{U}_l\right)\left(\mathbf{U}_j.\boldsymbol{\alpha}_k\right)\left(\boldsymbol{\alpha}_j.\boldsymbol{\alpha}_l\right)\\
&+ \frac{\lambda}{8}\sum_{j,k,l,m}\left(\mathbf{U}_j.\mathbf{U}_k\right)\left(\mathbf{U}_l.\mathbf{U}_m\right)\left(\boldsymbol{\alpha}_j.\boldsymbol{\alpha}_k\right)\left(\boldsymbol{\alpha}_l.\boldsymbol{\alpha}_m\right) + \frac{\mu}{4}\sum_{j,k,l,m}(\mathbf{U}_j.\mathbf{U}_k)(\mathbf{U}_l.\mathbf{U}_m)(\boldsymbol{\alpha}_j.\boldsymbol{\alpha}_m)(\boldsymbol{\alpha}_k.\boldsymbol{\alpha}_l)\\
&= \sum_{j,k}\mathbf{U}_j^t\left[\frac{\lambda}{2}\left(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k\right) + \frac{\mu}{2}\left(\boldsymbol{\alpha}_k \otimes \boldsymbol{\alpha}_j\right) + \frac{\mu}{2}\left(\boldsymbol{\alpha}_j.\boldsymbol{\alpha}_k\right)Id_3\right]\mathbf{U}_k && \text{(B.28)}\\
&+ \sum_{j,k,l}\left(\mathbf{U}_k.\mathbf{U}_l\right)\left\{\mathbf{U}_j.\left[\frac{\lambda}{2}\boldsymbol{\alpha}_j\ \left(\boldsymbol{\alpha}_k.\boldsymbol{\alpha}_l\right) + \frac{\mu}{2}\boldsymbol{\alpha}_l\ \left(\boldsymbol{\alpha}_j.\boldsymbol{\alpha}_k\right) + \frac{\mu}{2}\boldsymbol{\alpha}_k\ \left(\boldsymbol{\alpha}_j.\boldsymbol{\alpha}_l\right)\right]\right\}\\
&+ \sum_{j,k,l,m}\left(\mathbf{U}_j.\mathbf{U}_k\right)\left(\mathbf{U}_l.\mathbf{U}_m\right)\left[\frac{\lambda}{8}\left(\boldsymbol{\alpha}_j.\boldsymbol{\alpha}_k\right)\left(\boldsymbol{\alpha}_l.\boldsymbol{\alpha}_m\right) + \frac{\mu}{4}(\boldsymbol{\alpha}_j.\boldsymbol{\alpha}_m)(\boldsymbol{\alpha}_k.\boldsymbol{\alpha}_l)\right]
\end{aligned}$$

$$W = \underbrace{\sum_{j,k}\mathbf{U}_j^t\left[\mathcal{B}^{jk}\right]\mathbf{U}_k}_{W_1} + \underbrace{\sum_{j,k,l}\left(\mathbf{U}_j.\mathcal{C}^{jkl}\right)\left(\mathbf{U}_k.\mathbf{U}_l\right)}_{W_2} + \underbrace{\sum_{j,k,l,m}\mathcal{D}^{jklm}\left(\mathbf{U}_j.\mathbf{U}_k\right)\left(\mathbf{U}_l.\mathbf{U}_m\right)}_{W_3} \quad \text{(B.29)}$$

Afin de faciliter les calculs de dérivation de cette énergie, on peut remarquer les propriétés suivantes des matrices $\mathcal{B}^{jk}$, des vecteurs $\mathcal{C}^{jkl}$ et des réels $\mathcal{D}^{jklm}$:

$$\mathcal{B}^{jk} = \left(\mathcal{B}^{kj}\right)^t \tag{B.30}$$

$$\mathcal{C}^{jkl} = \mathcal{C}^{jlk} \tag{B.31}$$

$$\mathcal{D}^{jklm} = \mathcal{D}^{kjml} = \mathcal{D}^{lmjk} \tag{B.32}$$

La force interne élastique $\mathbf{F}^p$, exercée sur le sommet $\mathbf{P}_p$, s'exprime de la manière suivante :

$$\mathbf{F}^p = \left(\frac{\partial\, W}{\partial\, \mathbf{U}_p}\right)^t \tag{B.33}$$

Dérivation du premier terme :

$$\begin{aligned}
\frac{\partial\, W_1}{\partial\, \mathbf{U}_p} &= \sum_{j,k} \frac{\partial\, \mathbf{U}_j^t \left[\mathcal{B}^{jk}\right] \underline{\mathbf{U}_k}}{\partial\, \mathbf{U}_p} + \frac{\partial\, \mathbf{U}_k^t \left[\mathcal{B}^{jk}\right]^t \underline{\underline{\mathbf{U}_j}}}{\partial\, \mathbf{U}_p} \\
&= \sum_j \mathbf{U}_j^t \left[\mathcal{B}^{jp}\right] + \sum_k \mathbf{U}_k^t \left[\mathcal{B}^{pk}\right]^t \\
&= 2 \sum_j \mathbf{U}_j^t \left[\mathcal{B}^{jp}\right]
\end{aligned} \tag{B.34}$$

On retrouve bien l'expression de la force élastique pour un modèle linéaire :

$$\mathbf{F}_1^p = 2 \sum_j \left[\mathcal{B}^{pj}\right] \mathbf{U}_j \tag{B.35}$$

$$= \underbrace{2\left[\mathcal{B}^{pp}\right] \mathbf{U}_p}_{\text{Sommet p}} + \underbrace{2 \sum_{j \neq p} \left[\mathcal{B}^{pj}\right] \mathbf{U}_j}_{\text{Arêtes (p,j)}} \tag{B.36}$$

La dérivation du deuxième terme d'énergie donne :

$$\frac{\partial\, W_2}{\partial\, \mathbf{U}_p} = \sum_{j,k,l} \frac{\partial\, \mathbf{U}_j^t\, \mathcal{C}^{jkl}\, \mathbf{U}_k^t\, \underline{\mathbf{U}_l}}{\partial\, \mathbf{U}_p} + \frac{\partial\, \mathbf{U}_j^t\, \mathcal{C}^{jkl}\, \mathbf{U}_l^t\, \underline{\mathbf{U}_k}}{\partial\, \mathbf{U}_p} + \frac{\partial\, \mathbf{U}_k^t\, \mathbf{U}_l\, \left(\mathcal{C}^{jkl}\right)^t\, \underline{\underline{\mathbf{U}_j}}}{\partial\, \mathbf{U}_p} \tag{B.37}$$

$$= \sum_{j,k} \mathbf{U}_j^t\, \mathcal{C}^{jkp}\, \mathbf{U}_k^t + \sum_{j,l} \mathbf{U}_j^t\, \mathcal{C}^{jpl}\, \mathbf{U}_l^t + \sum_{k,l} \mathbf{U}_k^t\, \mathbf{U}_l\, \left(\mathcal{C}^{pkl}\right)^t \tag{B.38}$$

$$= \sum_{j,k} 2\left(\mathcal{C}^{jkp}\right)^t (\mathbf{U}_j \otimes \mathbf{U}_k) + (\mathbf{U}_j.\mathbf{U}_k)\left(\mathcal{C}^{pjk}\right)^t \tag{B.39}$$

D'où le deuxième terme de la force élastique :

$$\mathbf{F}_2^p = \sum_{j,k} 2\,(\mathbf{U}_k \otimes \mathbf{U}_j)\, \mathcal{C}^{jkp} + (\mathbf{U}_j.\mathbf{U}_k)\, \mathcal{C}^{pjk} \tag{B.40}$$

Cette force peut se décomposer de la manière suivante :

$$
\begin{aligned}
\mathbf{F}_2^p &= \underbrace{\left[2\,(\mathbf{U}_p \otimes \mathbf{U}_p) + (\mathbf{U}_p.\mathbf{U}_p)\,Id_3\right]\mathcal{C}^{ppp}}_{\text{Sommet p}} \\
&+ \underbrace{\sum_{j\neq p} 2\left[(\mathbf{U}_j \otimes \mathbf{U}_p) + (\mathbf{U}_j.\mathbf{U}_p)\,Id_3\right]\mathcal{C}^{ppj} + 2\,(\mathbf{U}_p \otimes \mathbf{U}_j)\,\mathcal{C}^{jpp} + 2\,(\mathbf{U}_j \otimes \mathbf{U}_j)\,\mathcal{C}^{jjp} + (\mathbf{U}_j.\mathbf{U}_j)\,\mathcal{C}^{pjj}}_{\text{Arêtes (p,j)}} \\
&+ \underbrace{2\sum_{\substack{j,k\neq p\\ j<k}} (\mathbf{U}_k \otimes \mathbf{U}_j)\,\mathcal{C}^{jkp} + (\mathbf{U}_j \otimes \mathbf{U}_k)\,\mathcal{C}^{kjp} + (\mathbf{U}_j.\mathbf{U}_k)\,\mathcal{C}^{pjk}}_{\text{Faces (p,j,k)}} \qquad \text{(B.41)}
\end{aligned}
$$

Enfin, le dernier terme d'énergie se dérive en :

$$
\frac{\partial\, W_3}{\partial\, \mathbf{U}_p} = 4\sum_{j,k,l} \mathcal{D}^{jklp}\, \mathbf{U}_j^t\, \mathbf{U}_k\, \mathbf{U}_l^t \qquad \text{(B.42)}
$$

et le troisième terme de la force vaut :

$$
\mathbf{F}_3^p = 4\sum_{j,k,l} \mathcal{D}^{jklp}\, \mathbf{U}_l\, \mathbf{U}_k^t\, \mathbf{U}_j \qquad \text{(B.43)}
$$

Ce terme peut se décomposer comme suit :

$$
\begin{aligned}
\mathbf{F}_3^p &= \underbrace{4\mathcal{D}^{pppp}\, \mathbf{U}_p\, \mathbf{U}_p^t\, \mathbf{U}_p}_{\text{Sommet p}} \\
&+ \underbrace{\begin{aligned} &4\sum_{j\neq p} \mathcal{D}^{jppp}\, \left(2\mathbf{U}_p\, \mathbf{U}_p^t\, \mathbf{U}_j + \mathbf{U}_j\, \mathbf{U}_p^t\, \mathbf{U}_p\right) + \mathcal{D}^{jjpp}\, \mathbf{U}_p\, \mathbf{U}_j^t\, \mathbf{U}_j \\ &\quad + \left(\mathcal{D}^{jpjp} + \mathcal{D}^{pjjp}\right) \mathbf{U}_j\, \mathbf{U}_j^t\, \mathbf{U}_p + \mathcal{D}^{jjjp}\, \mathbf{U}_j\, \mathbf{U}_j^t\, \mathbf{U}_j \end{aligned}}_{\text{Arêtes (p,j)}} \\
&+ \underbrace{\begin{aligned} &4\sum_{\substack{j,k\neq p\\ j<k}} \left(\mathcal{D}^{pjkp} + \mathcal{D}^{jpkp}\right) \left(\mathbf{U}_j\, \mathbf{U}_k^t\, \mathbf{U}_p + \mathbf{U}_k\, \mathbf{U}_j^t\, \mathbf{U}_p\right) + 2\mathcal{D}^{jkpp}\, \mathbf{U}_p\, \mathbf{U}_j^t\, \mathbf{U}_k \\ &\qquad + \left(\mathcal{D}^{kjjp} + \mathcal{D}^{jkjp}\right) \mathbf{U}_j\, \mathbf{U}_j^t\, \mathbf{U}_k + \mathcal{D}^{jjkp}\, \mathbf{U}_k\, \mathbf{U}_j^t\, \mathbf{U}_j \\ &\qquad + \left(\mathcal{D}^{jkkp} + \mathcal{D}^{kjkp}\right) \mathbf{U}_k\, \mathbf{U}_k^t\, \mathbf{U}_j + \mathcal{D}^{kkjp}\, \mathbf{U}_j\, \mathbf{U}_k^t\, \mathbf{U}_k \end{aligned}}_{\text{Faces (p,j,k)}} \\
&+ \underbrace{4\left[\left(\mathcal{D}^{jklp} + \mathcal{D}^{kjlp}\right) \mathbf{U}_l\, \mathbf{U}_j^t\, \mathbf{U}_k + \left(\mathcal{D}^{jlkp} + \mathcal{D}^{ljkp}\right) \mathbf{U}_k\, \mathbf{U}_j^t\, \mathbf{U}_l + \left(\mathcal{D}^{kljp} + \mathcal{D}^{lkjp}\right) \mathbf{U}_j\, \mathbf{U}_k^t\, \mathbf{U}_l\right]}_{\text{Tétraèdre (p,j,k,l)}}
\end{aligned}
$$

B.3 Le modèle de St Venant Kirchhoff transversalement isotrope

Nous avons traité, dans l'annexe B.1, le cas de l'élasticité linéaire transversalement isotrope. Nous généralisons ici ce concept à l'élasticité en grands déplacements de St Venant-Kirchhoff. On ne traitera que le terme lié à l'étirement, car c'est celui que nous utilisons le plus couramment, et aussi parce que le terme de cisaillement entraîne des calculs lourds et fastidieux.

Nous allons donc exprimer l'énergie élastique supplémentaire due à l'étirement dans la direction d'anisotropie :

$$\Delta W_{\text{Étir\_aniso1}} = \left(\frac{\Delta\lambda}{2} + \Delta\mu\right) I_4^2 \tag{B.44}$$
$$I_4 = \mathbf{a_0}^t \; E \; \mathbf{a}_0$$

Dans le cas de l'élasticité en grands déplacements, le tenseur de déformation E s'écrit :

$$E \;\; = \;\; \frac{1}{2}\sum_j \left(\mathbf{U}_j \otimes \alpha_j + \alpha_j \otimes \mathbf{U}_j\right) + \frac{1}{2}\sum_{j,k}(\mathbf{U}_j.\mathbf{U}_k)(\alpha_j \otimes \alpha_k) \tag{B.45}$$

On peut alors exprimer I_4,

$$\begin{aligned} I_4 &= \mathbf{a_0}^t \; E \; \mathbf{a}_0 \\ &= \frac{1}{2}\sum_j \big(\mathbf{a_0}^t \; \mathbf{U}_j \; \boldsymbol{\alpha}_j{}^t \; \mathbf{a}_0 + \mathbf{a_0}^t \; \boldsymbol{\alpha}_j{}^t \; \mathbf{U}_j \; \mathbf{a}_0\big) + \frac{1}{2}\sum_{j,k} \mathbf{a_0}^t \big(\mathbf{U}_j{}^t \; \mathbf{U}_k\big)\big(\boldsymbol{\alpha}_j \; \boldsymbol{\alpha}_k{}^t\big) \; \mathbf{a}_0 \\ &= \sum_j \mathbf{a_0}^t \big(\mathbf{U}_j \otimes \boldsymbol{\alpha}_j\big)\mathbf{a}_0 + \frac{1}{2}\sum_{j,k} \mathbf{a_0}^t \big(\mathbf{U}_j \; . \; \mathbf{U}_k\big)\big(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k\big) \; \mathbf{a}_0 \end{aligned} \tag{B.46}$$

puis on l'élève au carré pour obtenir :

$$\begin{aligned} I_4^2 = \; & \underbrace{\left(\sum_j \mathbf{a_0}^t \big(\mathbf{U}_j \otimes \boldsymbol{\alpha}_j\big)\mathbf{a}_0\right)^2}_{(1)} \; + \; \underbrace{\left(\sum_j \mathbf{a_0}^t \big(\mathbf{U}_j \otimes \boldsymbol{\alpha}_j\big)\mathbf{a}_0\right)\left(\sum_{j,k} \mathbf{a_0}^t \big(\mathbf{U}_j \; . \; \mathbf{U}_k\big)\big(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k\big) \; \mathbf{a}_0\right)}_{(2)} \\ + \; & \underbrace{\frac{1}{4}\left(\sum_{j,k} \mathbf{a_0}^t \big(\mathbf{U}_j \; . \; \mathbf{U}_k\big)\big(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k\big) \; \mathbf{a}_0\right)^2}_{(3)} \end{aligned} \tag{B.47}$$

où chacun des termes s'exprime :

$$\begin{aligned} (1) &= \sum_{j,k} \mathbf{a_0}^t \big(\mathbf{U}_j \otimes \boldsymbol{\alpha}_j\big)\mathbf{a}_0 \; \mathbf{a_0}^t \big(\mathbf{U}_k \otimes \boldsymbol{\alpha}_k\big)\mathbf{a}_0 \\ &= \sum_{j,k} \mathbf{U}_j^t \; \Big[\big(\mathbf{a}_0 \otimes \mathbf{a}_0\big) \; \big(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k\big) \; \big(\mathbf{a}_0 \otimes \mathbf{a}_0\big)\Big] \; \mathbf{U}_k \end{aligned} \tag{B.48}$$

On vérifie que l'on retrouve bien la composante linéaire de l'énergie transversalement isotrope d'étirement (annexe B.1).

$$
\begin{aligned}
(2) &= \sum_{j,k,l} \mathbf{a_0}^t \big(\mathbf{U}_j \otimes \boldsymbol{\alpha}_j\big) \mathbf{a}_0 \; \mathbf{a_0}^t \big(\mathbf{U}_k \,.\, \mathbf{U}_l\big)\big(\boldsymbol{\alpha}_k \otimes \boldsymbol{\alpha}_l\big) \; \mathbf{a}_0 \\
&= \sum_{j,k,l} \mathbf{U}_j{}^t \; \mathbf{a}_0 \; \boldsymbol{\alpha}_j{}^t \; \mathbf{a}_0 \; \mathbf{a}_0{}^t \; \boldsymbol{\alpha}_k \; \boldsymbol{\alpha}_l{}^t \; \mathbf{a}_0 \; \big(\mathbf{U}_k \,.\, \mathbf{U}_l\big) \\
&= \sum_{j,k,l} \mathbf{U}_j{}^t \; \underbrace{\big(\mathbf{a}_0 \otimes \mathbf{a}_0\big) \; \big(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k\big) \; \big(\mathbf{a}_0 \otimes \mathbf{a}_0\big) \; \boldsymbol{\alpha}_l}_{\mathcal{C}^{jkl}_{\mathbf{a}_0}} \; \big(\mathbf{U}_k \,.\, \mathbf{U}_l\big) \\
&= \sum_{j,k,l} \big(\mathbf{U}_j{}^t \,.\, \mathcal{C}^{jkl}_{\mathbf{a}_0}\big) \; \big(\mathbf{U}_k \,.\, \mathbf{U}_l\big)
\end{aligned}
\tag{B.49}
$$

$$
\begin{aligned}
(3) &= \frac{1}{4} \sum_{j,k,l,m} \mathbf{a}_0{}^t \big(\mathbf{U}_j \,.\, \mathbf{U}_k\big)\big(\boldsymbol{\alpha}_j \otimes \boldsymbol{\alpha}_k\big) \; \mathbf{a}_0 \; \mathbf{a}_0{}^t \big(\mathbf{U}_l \,.\, \mathbf{U}_m\big)\big(\boldsymbol{\alpha}_l \otimes \boldsymbol{\alpha}_m\big) \; \mathbf{a}_0 \\
&= \frac{1}{4} \sum_{j,k,l,m} \mathbf{a}_0{}^t \; \boldsymbol{\alpha}_j \; \boldsymbol{\alpha}_k{}^t \; \mathbf{a}_0 \; \mathbf{a}_0{}^t \; \boldsymbol{\alpha}_l \; \boldsymbol{\alpha}_m{}^t \; \mathbf{a}_0 \; \big(\mathbf{U}_j \,.\, \mathbf{U}_k\big) \; \big(\mathbf{U}_l \,.\, \mathbf{U}_m\big) \\
&= \frac{1}{4} \sum_{j,k,l,m} \underbrace{\big(\mathbf{a}_0 \,.\, \boldsymbol{\alpha}_j\big) \; \big(\mathbf{a}_0 \,.\, \boldsymbol{\alpha}_k\big) \; \big(\mathbf{a}_0 \,.\, \boldsymbol{\alpha}_l\big) \; \big(\mathbf{a}_0 \,.\, \boldsymbol{\alpha}_m\big)}_{\mathcal{D}^{jklm}_{\mathbf{a}_0}} \; \big(\mathbf{U}_j \,.\, \mathbf{U}_k\big) \; \big(\mathbf{U}_l \,.\, \mathbf{U}_m\big) \\
&= \frac{1}{4} \sum_{j,k,l,m} \mathcal{D}^{jklm}_{\mathbf{a}_0} \; \big(\mathbf{U}_j \,.\, \mathbf{U}_k\big) \; \big(\mathbf{U}_l \,.\, \mathbf{U}_m\big)
\end{aligned}
\tag{B.50}
$$

Publications

Picinbono, G., Lombardo, J., Delingette, H., et Ayache, N. (2001). « Improving Realism of a Surgery Simulator: Linear Anisotropic Elasticity, Complex Interactions and Force Extrapolation ». *The Journal of Visualization and Computer Animation*. accepted (also available as INRIA research report RR-4018)

Picinbono, G., Delingette, H., et Ayache, N. (2001). « Non-Linear Anisotropic Elasticity for Real-Time Surgery Simulation ». submitted to *Graphical Models*. (also available as INRIA research report RR-4028)

Picinbono, G., Delingette, H., et Ayache, N. (2001). « Modèle déformable élastique non-linéaire pour la simulation de chirurgie en temps réel ». submitted to *Comptes Rendus de l'Académie des Sciences*.

Picinbono, G., Delingette, H., et Ayache, N. (2001). « Non-Linear and Anisotropic Elastic Soft Tissue Models for Medical Simulation ». Dans *IEEE International Conference on Robotics and Automotion: ICRA 2001*, Seoul, Korea.

Picinbono, G., Delingette, H., et Ayache, N. (2000). « Real-Time Large Displacement Elasticity for Surgery Simulation: Non-Linear Tensor-Mass Model ». Dans *Third International Conference on Medical Robotics, Imaging And Computer Assisted Surgery: MICCAI 2000*, pages 643–652.

Picinbono, G., Lombardo, J., Delingette, H., et Ayache, N. (2000). « Anisotropic Elasticity and Force Extrapolation to Improve Realism of Surgery Simulation ». Dans *IEEE International Conference on Robotics and Automotion: ICRA 2000*, pages 596–602, San Francisco, CA.

Picinbono, G. et Lombardo, J.-C. (1999). « Extrapolation: a Solution for Force Feedback? ». Dans *International Scientific Workshop on Virtual Reality and Prototyping*, pages 117–125, Laval, France.

Bibliographie

AGLIATI, P.-O. (1999). « Animation temps-réel pour un simulateur d'opérations chirurgicales ». Master's thesis, Ecole Nationale Supérieure d'Informatique et de Mathématiques Appliquées de Grenoble : ENSIMAG. Stage effectué dans le laboratoire GRAVIR, au sein de l'équipr iMAGIS : `http://www-imagis.imag.fr/`.

ÇAKMAK, H. et KÜHNAPFEL, U. (2000). « Animation and Simulation Techniques for VR-Training Systems in Endoscopic Surgery ». Dans *Eurographics Workshop on Animation and Simulation '2000 (EGCAS '2000)*, pages 173–185, Interlaken, Switzerland. ISBN: 3-211-83549-0.

ASTLEY, O. et HAYWARD, V. (1997). « Real-Time Finite-Elements Simulation of General Visco-elastic Materials for Haptic Presentation ». Dans *IROS '97, IEEE/RJS Int. Conference on Intelligent Robots and Systems*.

AYACHE, N. (1998). « L'analyse automatique des images médicales, état de l'art et perspectives ». *Annales de l'Institut Pasteur*, 9(1):13–21.

AYACHE, N., COTIN, S., et DELINGETTE, H. « Dispositif électronique de traitement de données-image, pour la simulation du comportement déformable d'un objet ». 19-11-1997. Brevet FR9714506, France (Europe - USA, WOFR9802428).

AYACHE, N., COTIN, S., et DELINGETTE, H. (1997). « Surgery simulation with visual and haptic feedback ». Dans *Proceedings of the Eighth International Symposium of Robotics Research*, pages 311–316, Japon. Springer.

AYACHE, N., COTIN, S., DELINGETTE, H., CLEMENT, J.-M., MARESCAUX, J., et NORD, M. (1998). « Simulation of endoscopic surgery ». *Journal of Minimally Invasive Therapy and Allied Technologies (MITAT)*, 7(2):71–77.

AZAR, F. S., METAXAS, D. N., et SCHNALL, M. D. (2000). « A Finite Element Model of the Breast for Predicting Mechanical Deformations during Biopsy Procedures ». Dans *IEEE Workshop on Mathematical Methods in biomedical Image Analysis*, South Carolina.

BARAFF, D. (1990). « Curved surfaces and coherence for non-penetrating rigid-body simulation ». *Computer Graphics*, 24(4):19–28. Proceedings of SIGGRAPH'90.

BASDOGAN, C., HO, C., et SRINIVASAN, M. (1997). « A Ray-Based Haptic Rendering Technique for Displaying Shape and Texture of 3D Objects in Virtual Environments ». Dans *the Winter Annual Meeting of ASME'97, DSC-Vol. 61*, pages 77–84, Dallas, TX.

Basdogan, C., Ho, C., et Srinivasan, M. (1999). « Simulation of Tissue Cutting and Bleeding for Laparoscopic Surgery Using Auxiliary Surfaces ». Dans *Medicine Meets Virtual Reality (MMVR'7) Conference*, pages 38–44, San Francisco, CA.

Basdogan, C., Ho, C., Srinivasan, M. A., Small, S. D., et Dawson, S. L. (1998). « Force Interaction in Laparoscopic Simulation: Haptic Rendering Soft Tissues ». Dans *Medecine Meets Virtual Reality (MMVR'6)*, pages 28–31, San Diego CA.

Bathe, K.-L. (1982). *Finite Element Procedures in Engineering Analysis*. Prentice-Hall.

Baumann, R. (1997). « *Haptic Interface for Virtual Reality Based Laparoscopic Surgery Training Environment* ». PhD thesis, Ecole Polytechnique Fédérale de Lausanne (EPFL). Thèse N° 1734.

Baur, C., Guzzoni, D., et Georg, O. (1998). « A Virtual Reality and Force Feedback Based Endoscopic Surgery Simulator ». Dans *Medicine Meets Virtual Reality, MMVR'98*, pages 110–116, San Diego.

Berkley, J., Oppenheimer, P., Weghorts, S., Berg, D., Raugi, G., Haynor, D., Ganter, M., Brooking, C., et Turkiyyyah, G. (2000). « Creating Fast Finite Element Models from Medical Images ». Dans *Medicine Meets Virtual Reality 2000 (MMVR'8)*, pages 26–32, Newport Beach, California.

Berkley, J., Weghorts, S., Gladstone, H., Raugi, G., Berg, D., et Ganter, M. (1999). « Fast Finite Element Modeling for Surgical Simulation ». Dans *Medicine Meets Virtual Reality (MMVR'7) Conference*, pages 55–61, San Francisco, CA.

Bielser, D., Maiwald, V., et Gross, M. (1999). « Interactive Cuts through 3-Dimensional Soft Tissue ». Dans *Computer Graphics Forum, Eurographics'99*.

Borouchaki, H. et Frey, P. (1999). « Maillage géométrique des surface ». *Revue Européenne des Éléments Finis*, 8:47–75.

Bourguignon, D. et Cani, M.-P. (2000). « Controlling Anisotropy in Mass-Spring Systems ». Dans *Proceedings of the 11th Eurographics Workshop on Animation and Simulation*. Springer-Verlag.

Boux de Casson, F. (2000). « *Simulation dynamique de corps biologiques et changements de topologie interactifs* ». PhD thesis, Université de Savoix.

Boux de Casson, F. et Laugier, C. (1999). « Modeling the Dynamics of a Human Liver for a Minimally Invasive Surgery Simulator ». Dans *Second International Conference on Medical Image Computing and Computer Assisted Intervention - MICCAI'99*, pages 1156–1165, Cambridge UK.

Boux de Casson, F. et Laugier, C. (2000). « Simulating 2D Tearing Phenomena for Interactive Medical Surgery Simulators ». Dans *Proc. of Computer Animation*, Philadelphia, PA (US).

Bro-Nielsen, M. (1996). « *Medical Image Registration and Surgery Simulation* ». PhD thesis, Department of Mathematical Modeling (IMM), Technical University of Danmark.

BRO-NIELSEN, M. (1997). « Simulation techniques for minimally invasive surgery ». *Journal of Minimally Invasive Therapy and Allied Technologies (MITAT)*, 6:106–110.

BRO-NIELSEN, M. et COTIN, S. (1996). « Real-time Volumetric Deformable Models for Surgery Simulation using Finite Elements and Condensation ». Dans *Eurographics '96. ISSN 1067-7055*, pages 57–66. Blackwell Publishers.

BRO-NIELSEN, M., HELFRICK, D., GLASS, B., ZENG, X., et CONNACHER, H. (1998). « VR Simulation of Abdominal Trauma Surgery ». Dans *Medicine Meets Virtual Reality (MMVR'98)*, pages 117–123.

BRONSHTEIN, I. et SEMENDYAYEV, K. (1985). *Handbook of Mathematics*. Verlag Harri Deutsch, Thun and Frank/Mainfurt.

BROOKS, T. (1990). « Telerobotic response requirements ». Dans *IEEE International Conference on systems, Man and Cybernetics*, pages 113–120.

BURDEA, G., DESHPANDE, S., LIU, B., LANGRANA, N., et GOMEZ, D. (1997). « A Virtual Reality-based System for Hand Diagnosis and Rehabilitation ». *Presence - Teleoperation and Virtual Environments, MIT Press*, 6(2). invited article.

BURDEA, G., PATOUNAKIS, G., POPESCU, V., et WEISS, R. (1998). « Virtual Reality Training for the Diagnosis of Prostate Cancer ». Dans *IEEE International Symposium on Virtual Reality and Applications (VRAIS'98)*, pages 190–197, Atlanta, Georgia.

BURDEA, G., PATOUNAKIS, G., POPESCU, V., et WEISS, R. (1999). « Virtual Reality-based Training for the Diagnosis of Prostate Cancer ». *IEEE Transactions on Biomedical Engineering*, 46(10):1253–1260.

CACHIER, P. et PENNEC, X. (2000). « 3D Non-Rigid Registration by Gradient Descent on a Gaussian-Windowed Similarity Measure using Convolutions ». Dans *Proc. of MMBIA'00*, pages 182 – 189, Hilton Head Island, USA.
`http://www-sop.inria.fr/epidaure/personnel/Pascal.Cachier/publi.html`.

CACHIER, P. et REY, D. (2000). « Symmetrization of the Non-Rigid Registration Problem using Inversion-Invariant Energies: Application to Multiple Sclerosis ». Dans *Proc. of MICCAI'00*, pages 472 – 481, Pittsburgh, USA.
`http://www-sop.inria.fr/epidaure/personnel/Pascal.Cachier/publi.html`.

CARTER, F. et DAVIES, P. (1997). « Biomechanical and physical modelling of the handling characteristics of tissues and organs during surgical interventions ». Darpa progress reports, University of Dundee.

CARTER, F. J. (1998). « Biomechanical testing of intra-abdominal soft tissue ». Dans *International Workshop on Soft Tissue Deformation and Tissue Palpation*, Cambridge, MA.

CHEN, E. (1999). « Six Degree-of-Freedom Haptic System for Desktop Virtual Prototyping Applications ». Dans *International Scientific Workshop on Virtual Reality and Prototyping*, pages 97–106, Laval, France.

CHEN, Y., ZHU, Q.-H., et KAUFMAN, A. (1998). « Physically-based Animation of Volumetric Objects ». Dans *Computer Animation '98*, pages 154–160.

CHUI, C. K. et CHEN, G. (1987). *Kalman filtering with real-time applications.* Springer.

CIARLET, P. G. (1988). *Mathematical elasticity Vol. 1: Three-dimensional elasticity.* Elsevier Science Publishers B.V.

CIARLET, P. G. (1990). *Plates and Junctions in Elastic multi-Structures.* Springer-Verlag.

CIARLET, P. G. (1997). *Mathematical Elasticity*, volume 2 : Theory of plates. Elsevier Science Publishers B.V.

CINQUIN, P. et TROCCAZ, J. (1999). « Gestes médico-chirurgicaux assistés par ordinateur et télé-chirurgie ». *Technologie et Santé, Special issue "Santé et nouvelles technologies de l'information et de la communication"*, 36:59–67.

COHEN, A. et CHEN, E. (1999). « Six Degree-of-Freedom Haptic System for Desktop Virtual Prototyping Applications ». Dans *ASME Winter Annual Meeting, Dynamics Systems and Control, DSC*, volume 67, pages 401–402, Nashville, Tennessee.

COTIN, S. (1997). « *Modèle anatomiques déformables en temps-réel: Application à la simulation de chirurgie avec retour d'effort* ». PhD thesis, Université de Nice-Sophia Antipolis.

COTIN, S., DELINGETTE, H., et AYACHE, N. (1999). « Real-time elastic deformations of soft tissues for surgery simulation ». *IEEE Transactions On Visualization and Computer Graphics*, 5(1):62–73.

COTIN, S., DELINGETTE, H., et AYACHE, N. (2000). « A Hybrid Elastic Model for Real-Time Cutting, Deformations and Force Feedback for Surgery Training and Simulation ». *The Visual Computer*, 16(8):437–452. Also available as INRIA research report RR-3510.

COTIN, S., DELINGETTE, H., CLÉMENT, J., MARESCAUX, J., et AYACHE, N. (1997). « Simulation Active de Chirurgie Endoscopique ». *Revue Européenne BioMédicale (RBM)*, 19(5):167–172.

COVER, S. A., EZQUERRA, N. F., et O'BRIEN, J. F. (1993). « Interactively Deformable Models for Surgery Simulation ». *IEEE: Computer Graphics and Applications*, 13(6):68–75.

DAN, D. (1999). « *Caractérisation mécanique du foie humain en situation de choc* ». PhD thesis, Université Paris 7.

DAN, D. et FLAUD, P. (1998). « Elaboration d'une methode de caracterisation mecanique sur foie : demarche experimentale ». Dans *Archives of Physiology and Biochemistry, Societe de Biomecanique 23 ieme Congres.*

DAN, D., FLAUD, P., POTIER, P., et ROBIN, S. (2000). « Human liver caracterisation in compression and relaxation tests ». Dans *Biorheology 2000.*

D'AULIGNAC, D. et BALANIUK, R. (1999). « Providing reliable force-feedback for a virtual echographic exam of the human thigh ». Dans *Proc. of the PHANToM Users Group Workshop*, Boston, MA (US).

D'AULIGNAC, D., BALANIUK, R., et LAUGIER, C. (2000). « A Haptic Interface for a Virtual Exam of the Human Thigh ». Dans *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pages 2452–2457, San Francisco, CA (US).

D'AULIGNAC, D., LAUGIER, C., et CAVUSOGLU, M. (1999a). « Modelling the dynamics of a human thigh for a realistic echographic simulator with force feedback ». Dans *Second International Conference on Medical Image Computing and Computer Assisted Intervention - MICCAI'99*, pages 1191–1198, Cambridge, UK.

D'AULIGNAC, D., LAUGIER, C., et CAVUSOGLU, M. C. (1999b). « Towards a realistic echographic simulator with force feedback ». Dans *Proc. of the IEEE-RSJ Int. Conf. on Intelligent Robots and Systems*, pages 727–732, Kyongju (KR).

DAVATZIKOS, C. (1996). « Nonlinear Registration of Brain Images Using Deformable Models ». Dans *IEEE Workshop on Math. Methods in Biomedical Image Analysis*, pages 94–103, San Francisco.

DAVATZIKOS, C. (1997). « Spatial Transformation and Registration of Brain Images Using Elastically Deformable Models ». *Computer Vision and Image Understanding, Special Issue on Medical Imaging*, 66(2):207–222.

DAVIES, P. J., CARTER, F. J., ROXBURGH, D. C., et CUSCHIERI, A. (1999). « Mathematical modelling for keyhole surgery simulations: spleen capsule as an elastic membrane ». *The Journal of Theoretical Medicine*, 1:247–62.

DEBUNNE, G., DESBRUN, M., BARR, A., et CANI, M.-P. (1999). « Interactive multiresolution animation of deformable models ». Dans *10th Eurographics Workshop on Computer Animation and Simulation (CAS'99)*.

DEBUNNE, G., DESBRUN, M., CANI, M.-P., et BARR, A. (2000). « Adaptive Simulation of Soft Bodies in Real-Time ». Dans *Computer Animation*.

DELINGETTE, H. (1994). « *Modélisation, Déformation et Reconnaissance d'objets tridimensionnels a l'aide de maillages simplexes* ». PhD thesis, Ecole Centrale de Paris.

DELINGETTE, H., COTIN, S., et AYACHE, N. (1999). « A Hybrid Elastic Model allowing Real-Time Cutting, Deformations and Force-Feedback for Surgery Training and Simulation ». Dans *Computer Animation*, pages 70–81, Geneva, Switzerland. IEEE Computer Society.

DELP, S., LOAN, J., BASDOGAN, C., et ROSEN, J. (1997). « Surgical Simulation: An emerging technology for emergency medical training ». *Presence: Teleoperators and Virtual Environments (Special issue on Virtual Environments in Medicine), MIT Press*, 6(4):147–159.

DESBRUN, M. (1997). « *Modélisation et Animation de Matériaux Hautement Déformables en Synthèse d'Images* ». PhD thesis, Université Joseph Fourier.

DESBRUN, M. et GASCUEL, M.-P. (1995). « Animating Soft Substances with Implicit Surfaces ». Dans *SIGGRAPH'95*, Los Angeles, USA.

DESBRUN, M. et GASCUEL, M.-P. (1996). « Smoothed Particles: A new paradigm for animating highly deformable bodies ». Dans *6th Eurographics Workshop on Animation and Simulation'96*, Poitiers, France.

DESBRUN, M., SCHRÖDER, P., et BARR, A. (1999). « Interactive Animation of Structured Deformable Objects ». Dans *Graphics Interface '99*, Kingston, Canada.

DEUSSEN, O., KOBBELT, L., et TÜCKE, P. (1995). « Using Simulated Annealing to Obtain Good Nodal Approximations of Deformable Bodies ». Dans *Sixth Eurographics Workshop on Simulation and Animation*.

DINSMORE, M., LANGRANA, N., et BURDEA, G. (1997). « Virtual Reality Training Simulation for Palpation of Subsurface Tumors ». Dans *IEEE International Symposium on Virtual Reality and Applications (VRAIS'97)*, Albuquerque, New Mexico.

DUNCAN, J. et AYACHE, N. (2000). « Medical Image Analysis: Progress over two decades and the challenges ahead ». *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI*, 22(1):85–106.

FALOUTSOS, P., van de PANNE, M., et TERZOPOULOS, D. (1997). « Dynamic Free-Form Deformations for Animation Synthesis ». *IEEE Transaction on Visualization and Computer Graphics*, 3(4):201–214.

FERRANT, M., MACQ, B., et WARFIELD, S. (2000a). « Deformable modeling for characterizing biomedical shape changes ». Dans *DGCI 2000 - Discrete Geometry for Computer Imagery Conference*, Uppsala, Sweden. to appear.

FERRANT, M., WARFIELD, S., GUTTMANN, C., JOLESZ, F., et KIKINIS, R. (1999). « 3D Image matching using a finite element based elastic deformation model ». Dans *Second International Conference on Medical Image Computing and Computer Assisted Intervention - MICCAI'99 - Lecture Notes in Computer Sciences*, volume 1679, pages 202–209, Cambridge, UK.

FERRANT, M., WARFIELD, S., NABAVI, A., F.A.JOLESZ, et KIKINIS, R. (2000b). « Registration of 3D Intraoperative MR Images of the Brain Using a Finite Element Biomechanical Model ». Dans *Third International Conference on Medical Robotics, Imaging And Computer Assisted Surgery: MICCAI 2000*, pages 19–28.

FRANÇOIS, D., PINEAU, A., et ZAOUI, A. (1990). *Comportement mécanique des matériaux*. Hermes.

FUNG, Y. (1965). *Foundations of solid mechanics*. Prentice-Hall.

FUNG, Y. C. (1993). *Biomechanics - Mechanical Properties of Living Tissues*. Springer-Verlag, 2^{nd} edition.

GANOVELLI, F., CIGNONI, P., MONTANI, C., et SCOPIGNO, R. (2000a). « A Multiresolution Model for Soft Objects supporting interactive cuts and lacerations ». Dans *Computer Graphics Forum, Eurographics 2000 (to appear)*.

GANOVELLI, F., CIGNONI, P., MONTANI, C., et SCOPIGNO, R. (2000b). « Enabling Cuts on Multiresolution Representation ». Dans *CGI 2000 (N.Magnenat-Thalmann and D.Thalmann, eds.)*. (in press).

GANOVELLI, F., CIGNONI, P., et SCOPIGNO, R. (1999). « Introducing Multiresolution Representation in Deformable Object Modeling ». Dans *SCCG '99*, pages 149–158, Budmerice, Slovakia.

GARRIGUES, J. (1999a). « *Elasticité classique* ».
http://esm2.imt-mrs.fr/gar/elas.html.

GARRIGUES, J. (1999b). « *La méthode des éléments finis* ».
http://esm2.imt-mrs.fr/gar/ef.html.

GARRIGUES, J. (1999c). « *Éléments d'algèbre et d'analyse tensorielle à l'usage des mécaniciens* ».
http://esm2.imt-mrs.fr/gar/tenseurs.html.

GARRIGUES, J. (1999d). « *Mécanique des milieux continus* ».
http://esm2.imt-mrs.fr/gar/mmc.html.

GELDER, A. V. (1998). « Approximate Simulation of Elastic Membranes by Triangulated Spring Meshes ». *Journal of Graphics Tools: JGT*, 3(2):21–41.
http://www.acm.org/jgt/papers/VanGelder98/.

GELDER, A. V. et WILHELMS, J. (1997). « Simulation of Elastic Membranes and Soft Tissue with Triangulated Spring Meshes ». Technical Report UCSC-CRL-97-12, University of California, Santa Cruz, Jack Baskin School of Engineering.
ftp://ftp.cse.ucsc.edu/pub/tr/ucsc-crl-97-12.ps.Z.

GEORGE, P. (1997). « Improvement on Delaunay based 3D automatic mesh generator ». *Finite Elements in Analysis and Design*, 25:297–317.

GEORGE, P. (1999a). « Automatic 3D Mesh Generation conforming a Prescribed Size Map ». Dans *Numerical grid generation in computational Field Simulations*, pages 21–32, Greenwich, UK.

GEORGE, P. (1999b). « Tet meshing : Construction, Optimization and Adaptation ». Dans *8th Int. Meshing Roundtable*, pages 133–141, Lake Tahoe, CA.

GEORGE, P. et BOROUCHAHI, H. (1999). « Génération automatique de maillages tridimensionnels respectant une carte de taille ». *Revue Européenne des Éléments Finis , , p. .*, 7:339–363.

GIBSON, S. (1997). « 3D ChainMail: a fast algorithm for deforming volumetric objects ». Dans *1997 Symposium on Interactive 3D Graphics*, pages 149–154, Providence, RI, USA.

GIBSON, S., FYOCK, C., GRIMSON, E., KANADE, T., KIKINIS, R., LAUER, H., MCKENZIE, N., MOR, A., NAKAJIMA, S., OHKAMI, H., OSBORNE, R., SAMOSKY, J., et SAWADA, A. (1998). « Volumetric object modeling for surgical simulation ». *Medical Image Analysis*, 2(2):121–132.

GOTTSCHALK, S., GOPI, M., LIN, M., MANOCHA, D., et PATTEKAR, A. (1996). « OBBTree: A Hierarchical Structure for Rapid Interference Detection ». *Computer Graphics*. Proceedings of SIGGRAPH'96.

HATFIELD, G. A. (1996). « Conservation laws and canonical forms in the stroh formalism of anisotropic elasticity ». *Quarterly of Applied Mathematics*, 54:739–758.

HATFIELD, G. A. (1997). « Conservation laws in transversely isotropic linear elastic materials ». *Proceedings of the Royal Society of London A*, 453:1005–1017.

HIROTA, G., FISHER, S., et LIN., M. (2000). « Simulation of Non-penetrating Elastic Bodies Using Distance Fields ». Rapport technique, University of North Carolina at Chapel Hill, NC.

Ho, C., Basdogan, C., et Srinivasan, M. (1997). « Haptic Rendering: Point - and Ray - Based Interactions ». Dans *Second PHANToM Users Group Workshop, PUG'97*, Dedham, MA.

Hughes, T. J. (1987). *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*. Prentice-Hall, Inc., Enlewood Cliffs, New Jersey.

Humphrey, J. D., Strumpf, R. K., et Yin, F. C. P. (1990). « Determination of a Contitutive Relation for Passive Myocardium: I. A New Functional Form ». *ASME Journal of Biomechanical Engineering*, 112:333–339.

Humphrey, J. D. et Yin, F. C. P. (1987). « On Constitutive Relations and Finite Deformations of Passive Cardiac Tissue: I. A Pseudostrain-Energy Function ». *ASME Journal of Biomechanical Engineering*, 109:298–304.

Hunter, P. et Pullan, A. (1998). « *FEM/BEM Notes* ». Bioengineering Research Group, Department of Engineering Science, University of Auckland, New Zealand. `http://www.esc.auckland.ac.nz/Academic/Texts/FEM-BEM-notes.html`.

Hutchinson, D., Preston, M., et Hewitt, T. (1996). « Adaptive Refinement for Mass/Spring Simulation ». Dans *7th Eurographics Workshop on Animation and Simulation*, pages 31–45, Poitiers, France.

Jaillet, F. (1999). « *Contribution à la reconstruction et à l'animation d'objets déformables définis à partir de données structurées en sections* ». PhD thesis, Université Claude Bernard - Lyon I.

Jaillet, F., Shariat, B., et Vandorpe, D. (1997). « Deformable Volume Object Modeling with a Particle Based System for Medical Applications ». Dans *WSCG'97*, pages 192–201, Plzen (CZ), pp. , 1997.

Jambon, A., Dubois, P., et Karpf, S. (1997). « A Low-Cost Training Simulator for Initial Formation in Gynecologic Laparoscopy ». Dans *CVRMed'97*, pages 347–356, Grenoble, France.

Jambon, A., Querleu, D., Dubois, P., Chaillou, C., Meseure, P., Karpf, S., et Géron, C. (2000). « SPIC : Pedagogical Simulator for Gynecologic Laparoscopy ». Dans *8th Medecine Meets Virtual Reality Conference*, pages 139–145, Newport Beach.

James, D. L. et Pai, D. K. (1999). « ArtDefo: accurate real time deformable objects ». *Computer Graphics*, 33(Annual Conference Series):65–72. `http://www.acm.org/pubs/citations/proceedings/graph/311535/p65-james/`.

Kaiss, M. et Tallec, P. L. (1996). « La modélisation numérique du contact œil-trépan ». *Revue Européenne des éléments Finis*, 5(3):375–408.

Kardestuncer, H. (1987). *Finite element handbook*. Mc Graw-Hill, ISBN 0-07-033305-X.

Kauer, M., Vuskovic, V., et Dual, J. (1999). « In-Vivo Measurement of Elasto-Mechanical Properties of Organic Tissue ». Dans *European Medical & Biological Engineering Conference, EMBEC'99*, pages 270–271, Vienna, Austria.

Kühnapfel, U., Çakmak, H., et Maass, H. (1999). « 3D Modeling for Endoscopic Surgery ». Dans *IEEE Symposium on Simulation*, pages 22–32, Delft University, Delft, NL. ISBN: 90-804551-7-2.

Kühnapfel, U., Kuhn, C., Hübner, M., Krumm, H.-G., Maass, H., et Neisius, B. (1997). « The Karlsruhe Endoscopic Surgery Trainer as an example for Virtual Reality in Medical Education ». *Minimally Invasive Therapy and Allied Technologies (MITAT)*, 6:122–125.

Krissian, K. (2000). « *Traitement multi-échelle: application à l'imagerie médicale et à la détection tridimentionnelle de vaisseaux* ». PhD thesis, Université de Nice-Sophia Antipolis.

Kruse, R., Gebhardt, J., et Klawonn, F. (1994). *Foundations of Fuzzy Systems*. John Wiley & Sons, Inc., New York.

Kuhn, C., Kühnapfel, U., et Krumm, H.-G. (1996). « "Kalsruhe Endoscopic Surgery Trainer": A "Virtual Reality" based Training System for minimally Invasive Surgery ». Dans *Computer Assisted Surgery (CAR'96)*, Paris.

Kyriacou, S. K., Davatzikos, C., Zinreich, S. J., et Bryan, R. N. (1998). « Modeling Brain Pathology and Tissue Deformation Using a Finite Element Based Nonlinear Elastic Model ». *IEEE Transaction on Medical Imaging*.

Kyriacou, S. K., Davatzikos, C., Zinreich, S. J., et Bryan, R. N. (1999). « Nonlinear Elastic Registration of Brain Images with Tumor Pathology Using a Biomechanical Model ». *IEEE Transaction on Medical Imaging*, 18(7):580–592.

Lamy, D. et Chaillou, C. (1999). « Design, Implementation and Evaluation of an Haptic Interface for Surgical Gestures Training ». Dans *International Scientific Workshop on Virtual Reality and Prototyping*, pages 107–116, Laval, France.

Legat, V. (1999-2000). « *Introduction aux éléments finis. Notes de cours MECA 2120* ». Faculté des Sciences Appliquées, Université catholique de Louvain.

Lin, C. T. et Lee, C. S. G. (1996). *Neural Fuzzy Systems: A Neuro-Fuzzy Synergism to Intelligent Systems*. Prentice-Hall Inc., New Jersey.

Lin, M. et Canny, J. (1992). « Efficient collision detection for animation ». Dans *Third Eurographics Workshop on Animation and Simulation*, Cambridge, England.

Lin, M. et Gottschalk, S. (1998). « Collision Detection between geometric models: a survey ». Dans *IMA conference on Mathematics of Surfaces*.

Lombardo, J.-C., Cani, M.-P., et Neyret, F. (1999). « Real-time collision detection for virtual surgery ». Dans *Computer Animation*, Geneva Switzerland.

Lumsdaine, A., Lee, L.-Q., et Siek, J. (1998). « The Iterative Template Library ». `http://www.lsc.nd.edu/research/itl/`.

Lumsdaine, A. et Siek, J. (1998). « The Matrix Template Library ». `http://www.lsc.nd.edu/research/mtl/`.

Maass, H. et Kühnapfel, U. (1999). « Noninvasive Measurement of Elastic Properties of Living Tissue ». Dans *13th Internat. Congress on Computer Assisted Radiology and Surgery (CARS '99)*, pages 865–870, Paris, France.

Marescaux, J., Clément, J.-M., Tassetti, V., Koehl, C., Cotin, S., Russier, Y., Mutter, D., Delingette, H., et Ayache, N. (1998). « Virtual Reality Applied to Hepatic Surgery Simulation: The Next Revolution ». *Annals of Surgery*, 228(5):627–634.

Martelli, S., Joukhadar, A., Zaffagnini, S., et Marcacci, M. (1997). Development and validation of a fiber-based ACL model for surgical simulations. Dans Troccaz, J., Grimson, E., et Mösges, R., éditeurs, *CVRMed-MRCAS'97*, Lecture Notes in Computer Science, pages 644–652. Springer-Verlag.

Martin, J., Pentland, A., Sclaroff, S., et Kikinis, R. (1998). « Characterization of Neuropathological Shape Deformations ». *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(2):97–112.

Massie, T. et Salisbury, J. (1994). « The PHANToM Haptic Interface: A Device for Probing Virtual Objects ». Dans *ASME Haptic Interfaces for Virtual Environment and Teleoperator Systems 1994, In Dynamic Systems and Control 1994*, volume 1, pages 295–301, Chicago.

Maurel, W., Wu, Y., Thalmann, N. M., et Thalmann, D. (1998). *Biomechanical Models for Soft Tissue Simulation*. Springer-Verlag.

Melvin, J. W., Stalnaker, R. L., et Roberts, S. H. (1973). « Impact Injury Mechanisms in Abdominal Organs ». *SAE Transactions 800098*, pages 115–126.

Meseure, P. (1997). « *Modélisation de corps deformables pour la simulation d'actes chirurgicaux* ». PhD thesis, Université des sciences et technologies de Lille.

Meseure, P. et Chaillou, C. (1997a). « Deformable Body simulation with Adaptive Subdivision and Cutting ». Dans *WSCG'97*, pages 361–370, Plzen, Republique Tchèque.

Meseure, P. et Chaillou, C. (1997b). « Détection de collisions entre modèles polyédriques : quelques propositions ». Dans *5ème séminaire du groupe de travail Animation et Simulation*, pages 91–100, Reims, France.

Meseure, P. et Chaillou, C. (2000). « A deformable body model for surgical simulation ». *The Journal of Visulization and Computer Animation*, 11:197–208.

Meseure, P., Hilde, L., et Chaillou, C. (1998). « Accélération de la détection de collisions entre corps rigides et déformables ». Dans *6èmes journées du Groupe de Travail Réalité Virtuelle, PRC-GDR Communication Homme-Machine*, pages 167–174, Paris, France.

Miga, M., Staubert, A., Paulsen, K., Kennedy, F., Tronnier, V., Roberts, D., Hartov, A., Platenik, L., et Lunn, K. (2000). « Model-Updated Image-Guided Neurosurgery: Preliminary Analysis Using Intraoperative MR ». Dans *Third International Conference on Medical Robotics, Imaging And Computer Assisted Surgery: MICCAI 2000*, pages 115–124.

Miga, M. I., Paulsen, K. D., Kennedy, F. E., Hoopes, P. J., Hartov, A., et Roberts, D. W. (1997). « A 3D Brain Deformation Model Experiencing Comparable Surgical Loads ». Dans *19th Annual International Conference of the IEEE Engineering in Medecine and Biology Society*, pages 773–776.

Miga, M. I., Paulsen, K. D., Kennedy, F. E., Hoopes, P. J., Hartov, A., et Roberts, D. W. (1998a). « Modeling Surgical Loads to Account for Sbsurface Tissue Deformation During

Stereotactic Neurosurgery ». Dans *1998 Biomedical Optics Symposium of the Internation al Society for Optical Engineers*.

MIGA, M. I., PAULSEN, K. D., KENNEDY, F. E., HOOPES, P. J., HARTOV, A., et ROBERTS, D. W. (1998b). « Quantification of a 3D Brain Deformable Model Experiencing a Temporal Mass Expansion ». Dans *24th IEEE Notheast Bioengineering Conference*, pages 68–71.

MILLER, K. (2000). « Constitutive Model of Abdominal Organs ». *Journal of Biomechanics*, 33(3):367–373.

MONAGHAN, J. J. (1992). « Smoothed Particules Hydrodynamics ». *Annual Reviews of Astronomy and Astrophysics*, 30(543).

MONTAGNAT, J. (1999). « *Modèles déformables pour la segmentation et la modélisation d'images médicales 3D et 4D* ». PhD thesis, Université de Nice-Sophia Antipolis.
`http://www.inria.fr/RRRT/TU-0612.html`.

MONTAGNAT, J. et DELINGETTE, H. (1998). « Globally constrained deformable models for 3D object reconstruction ». *Signal Processing*, 71(2):173–186. ftp://ftp-sop.inria.fr/epidaure/Publications/Montagnat/SPAIJ.ps.gz.

MOR, A. (1998). « 5 DOF Force Feedback Using the 3DOF Phantom and a 2DOF Device ». Dans *Proceedings of the Third PHANTOM Users Group Workhop*, volume 1643. MIT AI Laboratory.

MOR, A. B. et KANADE, T. (2000). « Modifying Soft Tissue Models: Progressive Cutting with Minimal New Element Creation ». Dans *Third International Conference on Medical Robotics, Imaging And Computer Assisted Surgery: MICCAI 2000*, pages 598–607, Pittsburgh, PA, USA.

MURTA, A. et MILLER, J. (1999). « Modelling and Rendering Liquids in Motion ». Dans *WSCG'99*, pages 194–201, Plzen-Bory, Czech Republic. ISBN: 80-7082-490-5.

NAUCK, D., KLAWONN, F., et KRUSE, R. (1997). *Foundations of Neuro-Fuzzy Systems*. John Wiley & Sons Inc., New York.

NEDEL, L. et THALMANN, D. (1998a). « Modeling and Deformation of Human Body using an Anatomy-Based Approach ». Dans *Proc. Computer Animation '98*, pages 34–40, Philadelphia. IEEE Computer Society Press.

NEDEL, L. et THALMANN, D. (1998b). « Real Time Muscle Deformations Using Mass-Spring Systems ». Dans *Proc. CGI '98, IEEE Computer Society Press*.

NEUMANN, P., SADLER, L., et M.D., J. G. (1998). « Virtual Reality Vitrectomy Simulator ». Dans WELLS, W. M. et OTHERS, éditeurs, *First International Conference on Medical Image Computing and Computer-Assisted Intervention MICCAI'98*, numéro 1496 dans Lecture Notes in Computer Science, pages 910–917, Cambridge MA, USA. Springer Verlag.

NEUMANN, P. F. (2000a). « Near Real-Time Cutting ». Dans *SIGGRAPH Visual Proceedings*. Technical Sketch.

NEUMANN, P. F. (2000b). « *Virtual Reality Vitrectomy Simulator* ». PhD thesis, University of Illinois, Chicago, Illinois.
`http://www.bvis.uic.edu/paul/projects/eye/Papers/`.

NEYRET, F. et CANI, M.-P. (1999). « Pattern-based texturing revisited ». Dans *SIGGRAPH'99*, Los Angeles CA.

O'BRIEN, J. et HODGINS, J. (1999). « Graphical Modeling and Animation of Brittle Fracture ». Dans *Computer Graphics, Annual Conference Series, SIGGRAPH'99*, pages 137–146, Los Angeles, California.

ODEN, J. (1967). *Mechanics of elastic structures.* McGraw-Hill.

OGDEN, R. (1984). *Non-linear elastic deformation.* Ellis Horwood Limited.

OPPENHEIMER, P. et WEGHORTS, S. (1999). « Immersive Surgical Robotic Interfaces ». Dans *Medicine Meets Virtual Reality (MMVR'7) Conference*, pages 242–248, San Francisco, CA.

PAPADEMETRIS, X. (2000). « *Estimation of 3D Left Venticular Deformation from Medical IMages Using Biomechanical Models* ». PhD thesis, Yale University.

PAPADEMETRIS, X., SHI, P., DIONE, D. P., SINUSAS, A. J., CONSTABLE, R. T., et DUNCAN, J. S. (1999). « Recovery of Soft Tissue Object Deformation from 3D Image Sequences Using Biomechanical Models ». Dans *XVI-th International Conference on Information Processing In Medical Imaging, IPMI'99*, pages 352–357, Visegrád, Hungary.

PAULSEN, K. D., MIGA, M. I., KENNEDY, F. E., HOOPES, P. J., HARTOV, A., et ROBERTS, W. (1999). « A Computational Model for Tracking Subsurface Tissue Deformation During Stereotactic Neurosurgery ». *IEEE Transactions on Biomedical Engineering*, 46(2):213–225.

PICINBONO, G., DELINGETTE, H., et AYACHE, N. (2000a). « Non-Linear Anisotropic Elasticity for Real-Time Surgery Simulation ». Rapport technique RR-4028, INRIA. `http://www-sop.inria.fr/rapports/sophia/RR-4028.html`, submitted to Graphical Models.

PICINBONO, G., DELINGETTE, H., et AYACHE, N. (2000b). « Real-Time Large Displacement Elasticity for Surgery Simulation: Non-Linear Tensor-Mass Model ». Dans *Third International Conference on Medical Robotics, Imaging And Computer Assisted Surgery: MICCAI 2000*, pages 643–652.

PICINBONO, G., DELINGETTE, H., et AYACHE, N. (2001a). « Non-Linear and Anisotropic Elastic Soft Tissue Models for Medical Simulation ». Dans *IEEE International Conference on Robotics and Automotion: ICRA 2001*, Seoul, Korea.

PICINBONO, G., LOMBARDO, J., DELINGETTE, H., et AYACHE, N. (2000c). « Anisotropic Elasticity and Force Extrapolation to Improve Realism of Surgery Simulation ». Dans *IEEE International Conference on Robotics and Automotion: ICRA 2000*, pages 596–602, San Francisco, CA.

PICINBONO, G., LOMBARDO, J., DELINGETTE, H., et AYACHE, N. (2001b). « Improving Realism of a Surgery Simulator: Linear Anisotropic Elasticity, Complex Interactions and Force Extrapolation ». *The Journal of Visualization and Computer Animation.* accepted (also available as INRIA research report RR-4018) `http://www-sop.inria.fr/rapports/sophia/RR-4018.html`.

PICINBONO, G. et LOMBARDO, J.-C. (1999). « Extrapolation: a Solution for Force Feedback? ».

Dans *International Scientific Workshop on Virtual Reality and Prototyping*, pages 117–125, Laval, France.

Popescu, V., Burdea, G., et Bouzit, M. (1999). « Virtual Reality Simulation Modeling for a Haptic Glove ». Dans *Computer Animation'99 Conference*, pages 195–200, Geneva, Switzerland.

Popescu, V., Burdea, G., Bouzit, M., Girone, M., et Hentz, V. (2000). « Orthopedic Telerehabilitation with Virtual Force Feedback ». *IEEE Transactions on Information Technology in Biomedicine*, 4(1):45–51.

Péquignot, J. (1999). « Adaptation dynamique de maillages éléments finis tétraédriques ». Rapport technique, INRIA. Rapport de projet ESSI3 - CSI.

Puso, M. A. et Weiss, J. A. (1998). « Finite Element Implementation of Anisotropic Quasi-linear Viscoelasticity Using a Discrete Spectrum Approximation ». *ASME Journal of Biomechanical Engineering*, 120(1).

Rabbitt, R. (1997). « *Vectors, Tensors and Coordinate Systems* ». Department of Bioengineering, University of Utah.
`http://www.bioen.utah.edu/faculty/RDR/course_notes/directory.html`.

Radetzky, A., Nürnberger, A., et Pretschner, D. (1998). « The Simulation of Elastic Tissues in Virtual Medicine using Neuro-Fuzzy Systems ». Dans *Medical Imaging 98: Image Display*, pages 399–409.

Radetzky, A., Rudolph, M., Stefanon, W., Starkie, S., et andL.M. Auer, B. D. (2000). « Simulating Minimally Invasive Neurosurgical Interventions Using an Active Manipulator ». Dans *Third International Conference on Medical Robotics, Imaging And Computer Assisted Surgery: MICCAI 2000*, pages 578–587.

Rey, D., Subsol, G., Delingette, H., et Ayache, N. (1999). « Automatic Detection and Segmentation of Evolving Processes in 3D Medical Images: Application to Multiple Sclerosis ». Dans Kuba, A., Sámal, M., et Todd-Pokropek, A., éditeurs, *Information Processing in Medical Imaging, IPMI'99*, volume 1613 de *Lecture Notes in Computer Science*, pages 154–167, Visegrád, Hungary. Springer.
`http://link.springer.de/link/service/series/0558/bibs/1613/16130154.htm`.

Rhomberg, A., Brechbühler, C., Székely, G., et Tröster, G. (1999). « A Parallel Architecture for Interactive FEM Computations in a Surgery Simulator ». Dans *Proceedings of the Parallel Computing conference, ParCo99*, Delft, The Netherlands. The non-profit foundation ParCo Conferences. Accepted for publication.

Roche, A., Malandain, G., et Ayache, N. (2000). « Unifying Maximum Likelihood Approaches in Medical Image Registration ». *International Journal of Imaging Systems and Technology*, 11:71–80.

Sachdeva, A. K., Blair, P. G., et Hanane, D. M. « The Use of State-of-the-Art Simulations in Contemporary Surgical and Medical Education ».
`http://www.mentice.com/news/art_01.html`.

Salisbury, J. et Srinivasan, M., éditeurs (1996). *The First PHANToM Users Group Workshop (PUG'96)*, MIT Endicott House, Cambridge, MA. AI Lab Technical Report No. 1596 and RLE Technical Report No. 612, MIT, Dec 1996,
http://www.ai.mit.edu/conferences/pug/.

Salisbury, J. et Srinivasan, M., éditeurs (1997a). *The Second PHANToM Users Group Workshop (PUG'97)*, MIT Endicott House, Cambridge, MA. AI Lab Technical Report No. 1617 and RLE Technical Report No. 618, MIT, Dec 1997
http://www.ai.mit.edu/conferences/pug/.

Salisbury, J. et Srinivasan, M., éditeurs (1998). *The Third PHANToM Users Group Workshop (PUG'98)*, MIT Endicott House, Cambridge, MA. AI Lab Technical Report No. 1643 and RLE Technical Report No. 624, MIT, December 1998
http://www.ai.mit.edu/conferences/pug/.

Salisbury, J. et Srinivasan, M., éditeurs (1999). *The Fourth PHANToM Users Group Workshop (PUG'99)*, MIT Endicott House, Cambridge, MA. AI Lab Technical Report No. 1675 and RLE Technical Report No. 633, MIT, November 1999
http://www.ai.mit.edu/conferences/pug/.

Salisbury, J. et Srinivasan, M., éditeurs (2000). *The Fifth PHANToM Users Group Workshop (PUG'00)*, Aspen, Colorado.
http://www.cs.sandia.gov/SEL/conference/pug00/pug00.htm.

Salisbury, J. K. et Srinivasan, M. A. (1997b). « Phantom-Based Haptic Interaction with Virtual Objects ». *IEEE Computer Graphics and Applications*, 17(5):6–10.

Schill, M. A., Gibson, S., Bender, H.-J., et Manner, R. (1998). « Biomechanical Simulation of the Vitreous Humor in the Eye Using Enhanced ChainMail Algorithm ». Dans Wells, W. M. et others, éditeurs, *First International Conference on Medical Image Computing and Computer-Assisted Intervention MICCAI'98*, pages 679–687, Cambridge MA, USA. Springer Verlag.

Shawki, T. (1996). « *Lecture Notes on Fundamantals of Solid Mechanics* ». Department of Theorical and Applied Mechanics, University of Illinois.
http://www.tam.uiuc.edu/Faculty/Shawki.html.

Soler, L. (1998). « *Une nouvelle méthode de segmentation des structures anatomiques et pathologiques : application aux angioscanners 3D du foie pour la planification chirurgicale* ». PhD thesis, Université d'Orsay.

Soler, L., Clement, J.-M., Koehl, C., Delingette, H., Malandain, G., Ayache, N., Dourthe, O., et Marescaux, J. (2000a). « An automatic virtual patient reconstruction from CT-scans for Hepatic Surgical planning ». Dans *Medical Meet on Virtual Reality 2000*, pages 316–322, Los Angeles.

Soler, L., Delingette, H., Malandain, G., Montagnat, J., Ayache, N., Clement, J.-M., Koehl, C., Dourthe, O., Mutter, D., et Marescaux, J. (2000b). « A fully automatic anatomical, pathological and functionnal segmentation from CT-scans for Hepatic Surgery ». Dans *Medical Imaging 2000*, San Diego.

SPENCER, A. (1972). *Deformations of Fibre-Reinforced Materials*. Clarendon Press, Oxfordn.

SPENCER, A. (1984). *Continuum Theory of Fiber-Reinforced Composites*. Springer-Verlag, New York.

SRINIVASAN, M. et BASDOGAN, C. (1997). « Haptics in Virtual Environments: Taxonomy, Research Status, and Challenges ». *Computers and Graphics (Special issue on "Haptic Displays in Virtual Environments")*, 21(4):393–404.

SUZUKI, N., HATTORI, A., TAKATSU, A., KUMANO, T., IKEMOTO, A., ADACHI, Y., et UCHIYAMA, A. (1998). « Virtual Surgery System Using Deformable Organ Models and Force Feedback System with Three Fingers ». Dans WELLS, W. M. et OTHERS, éditeurs, *First International Conference on Medical Image Computing and Computer-Assisted Intervention MICCAI'98*, pages 397–403, Cambridge MA, USA. Springer Verlag.

SZÉKELY, G., BAJKA, M., BRECHBUHLER, C., DUAL, J., ENZLER, R., HALLER, U., HUG, J., HUTTER, R., IRONMONGER, N., KAUER, M., MEIER, V., NIEDERER, P., RHOMBERG, A., SCHMID, P., SCHWEITZER, G., THALER, M., VUSKOVIC, V., et TROSTER, G. (2000a). « Virtual Reality Based Simulation of Endoscopic Surgery ». *Presence, MIT press*, 9(3):310–333.

SZÉKELY, G., BRECHBÜHLER, C., HUTTER, R., RHOMBERG, A., et SCHMID, P. (1998). « Modelling of Soft Tissue Deformation for Laparoscopic Surgery Simulation ». Dans WELLS, W. M. et OTHERS, éditeurs, *First International Conference on Medical Image Computing and Computer-Assisted Intervention MICCAI'98*, numéro 1496 dans Lecture Notes in Computer Science, pages 550–561, Cambridge MA, USA. Springer Verlag.

SZÉKELY, G., BRECHBÜHLER, C., HUTTER, R., RHOMBERG, A., et SCHMID, P. (2000b). « Modelling of soft tissue deformation for laparoscopic surgery simulation ». *Medical Image Analysis*, 4:57–66.

TERZOPOULOS, D. et FLEICHER, K. (1988). « Deformable Models ». *The Visual Computer*, 4:306–331.

TERZOPOULOS, D., PLATT, J., BARR, A., et FLEISCHER, K. (1987). « Elastically Deformable Models ». *Computer Graphics*, 21(4):205–214. Proceedings of ACM SIGGRAPH'87 Conference, Anaheim, CA.

TSAP, L., D.B.GOLDGOF, SARKAR, S., et HUANG, W.-C. (1998). « Efficient Nonlinear Finite Element Modeling of Nonrigid Objects via Optimization of Mesh Models ». *Computer Vision and Image Understanding*, 69(3):330–350.

VERONDA, D. R. et WESTMANN, R. A. (1970). « Mechanical Characterization of Skin-Finite Deformations ». *Journal of Biomechanics*, 3:111–124.

VIDRASCU, M. (1999). « Simulation of the deformations of the liver by domain decomposition ». Dans *mini-symposim Decomposition de domaines de USNCCM99*, Boulder.

VUSKOVIC, V., BLASER, R., et SPIGA, A. (1999a). « Vision-Based Device for In-Vivo Measurement of Elasto-Mechanical Properties of Soft Tissues ». Dans *Euromech 406 Colloquium: Image Processing Methods in Applied Mechanics*, pages 227–230, Warsaw, Poland.

VUSKOVIC, V., KAUER, M., DUAL, J., et BAJKA, M. (1999b). « Method and device for in-vivo measurement of elasto-mechanical properties of soft biological tissues, ». *Machine Graphics & Vision International Journal*, 8(4).

VUSKOVIC, V., KAUER, M., SZÉKELY, G., et REIDY, M. (2000). « Realistic Force Feedback for Virtual Reality Based Diagnostic Surgery Simulators ». Dans *IEEE International Conference on Robotics and Automotion: ICRA 2000*, pages 1592–1598, San Francisco, CA.

WEISS, J. A. (1995). « A model for the material behavior of ligaments and tendons ». *Proc ASME Winter Annual Meeting*, BED-15:97–98.

WEISS, J. A., GARDINER, J. C., et QUAPP, K. M. (1995). « Material Models for the Study of Tissues Mechanics ». *Proc International conference on Pelvic and Lower Extrimity Injuries*, pages 249–261. Wash DC.

WEISS, J. A., MAKER, B. N., et GOVINDJEE, S. (1996). « Finite element implementation of incompressible, transversely isotropic hyper-elasticity ». *Computer Methods in Applied Mechanics and Engineering*, 135:107–128. U.C. Berkely Report UCB/SEMM-95/07.

WEISSTEIN, E. (2000). « MathWorld web site ».
`http://mathworld.wolfram.com/`.

WILHELMS, J. et GELDER, A. V. (1997). « Anatomically Based Modeling ». *Computer Graphics*, 31(Annual Conference Series):173–180.
`http://www.acm.org:80/pubs/citations/proceedings/graph/258734/p173-wilhelms/`.

WILHELMS, J. et VANGELDER, A. (1997). « Anatomically Based Modeling ». Technical Report UCSC-CRL-97-10, University of California, Santa Cruz, Jack Baskin School of Engineering.
`ftp://ftp.cse.ucsc.edu/pub/tr/ucsc-crl-97-10.ps.gz`.

ZHU, Q.-H., CHEN, Y., et KAUFMAN, A. (1998). « Real-time Biomechanically-based Muscle Volume Deformation using FEM ». Dans *Eurographics'98*.

ZHUANG, Y. (2000). « *Real-time Simulation of Physically-Realistic Global Deformations* ». PhD thesis, Department of Electrical Engineering and Computer Science, UC Berkeley.
`http://www.cs.berkeley.edu:80/~yzhuang/publications.html`.

ZHUANG, Y. et CANNY, J. (1999). « Real-time Simulation of Physically Realistic Global Deformation ». Dans *IEEE Vis'99 Late Breaking Hot Tropics*, Los Angeles, California.

ZHUANG, Y. et CANNY, J. (2000). « Haptic Interaction with Global Deformations ». Dans *IEEE International Conference on Robotics and Automotion: ICRA 2000*, pages 2428–2433, San Francisco, CA.

ZIENKIEWICZ, O. (1977). *The finite element method*. Mc Graw-Hill, London, 3^{rd} edition.

Index des références

– F –

– G –

– H –

– I –

– J –

– K –

– L –

– M –

– N –

– O –

– P –

– R –

– S –

Modèles géométriques et physiques pour la simulation d'interventions chirurgicales

Dans cette thèse, nous proposons un ensemble d'outils nécessaires à l'élaboration d'un simulateur de chirurgie. Dans un premier temps, nous définissons plusieurs modèles déformables physiques temps réels permettant de simuler les déformations et la découpe d'organes du corps humain. Ces modèles s'appuient sur la théorie de l'élasticité et la méthode des éléments finis. Nous avons tout d'abord travaillé sur l'enrichissement du modèle élastique linéaire en le généralisant au cas des matériaux dont le comportement est anisotrope, soit en raison de la présence de fibres (muscles, tendons), soit parce qu'ils sont entourés d'une peau (capsule de Glisson pour le foie). Cependant, la principale limitation de l'élasticité linéaire est de n'être valable que dans l'hypothèse de petits déplacements. Nous proposons donc un nouveau modèle déformable mettant en œuvre l'élasticité non-linéaire de St Venant-Kirchhoff, qui reste valable pour les grands déplacements. Après avoir étendu ce modèle aux matériaux anisotropes, nous proposons plusieurs méthodes d'optimisation des calculs, soit en utilisant un algorithme adaptatif qui combine les modèles linéaires et non-linéaires, soit à partir d'une nouvelle formulation. La seconde partie de ces travaux porte sur la simulation des interactions entre les instruments chirurgicaux et les organes virtuels. Pour cela, nous avons modélisé les contacts, ainsi que certaines actions spécifiques comme le glissement, la préhension et la découpe. De plus, nous nous sommes intéressés aux problèmes liés à l'utilisation d'interfaces à retour d'effort.

Mots clés : *Simulation de chirurgie, modèle déformable, élasticité, anisotropie, méthode des éléments finis, découpe, retour d'effort, temps réel.*

Geometrical and physical models for surgery simulation

In this thesis, we propose a set of tools necessary for the development of a surgery simulator. First, we define several real-time physically based deformable models allowing us to simulate the deformations and the cutting of human organs. These models are based on the theory of elasticity and the finite element method. We began with the improvement of the linear elastic model by generalizing it to materials having an anisotropic behavior, either because they contain fibers (muscles, tendons), or because they are surrounded by a skin (liver capsule). Nevertheless, the main shortcoming of linear elasticity is that it is only valid under the hypothesis of small displacements. Thus, we propose a new deformable model implementing the St Venant-Kirchhoff non-linear elasticity, which is also valid for large displacements. After having extended this model to anisotropic materials, we propose two optimization methods, either using an adaptive algorithm which combines linear and non-linear models, or implementing a new formulation. The second part of this work is dedicated to the simulation of the interactions between the surgical tools and the virtual organs. For this, we have modelized contacts and some special gestures like sliding, gripping, or cutting. Furthermore, we have addressed the problems related to the use of force feedback devices.

Keywords: *Surgery simulation, deformable model, elasticity, anisotropy, finite element method, cutting, force feedback, real-time.*