HAL
open science

Chirurgie virtuelle : modélisation temps réel des tissus mous, interactions et système haptique dédié

Safwan Chendeb

To cite this version:

Safwan Chendeb. Chirurgie virtuelle : modélisation temps réel des tissus mous, interactions et système haptique dédié. Mathématiques [math]. École Nationale Supérieure des Mines de Paris, 2007. Français. NNT : 2007ENMP1614 . pastel-00005287

HAL Id: pastel-00005287

https://pastel.hal.science/pastel-00005287

Submitted on 23 Jul 2009

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Collège doctoral

ED n° 431 : Information, communication, modélisation et simulation

N° attribué par la bibliothèque
|__|__|__|__|__|__|__|__|__|__|

T H E S E

pour obtenir le grade de
Docteur de l'Ecole des Mines de Paris
Spécialité "Informatique temps réel - Robotique - Automatique"

présentée et soutenue publiquement par
Safwan CHENDEB

le 20 décembre 2007

CHIRURGIE VIRTUELLE
MODELISATION TEMPS REEL DES TISSUS MOUS, INTERACTIONS ET SYSTEME HAPTIQUE DEDIE

Devant le jury composé de :

M. Christophe Chaillou	Professeur, Université des Sciences et Technologies de Lille	Rapporteur
M. Etienne Dombre	Directeur de Recherches, LIRMM – CNRS, Montpellier	Rapporteur
M. Serge Couvet	Ingénieur de Recherche, Thales Training et Simulation	Examinateur
M. Claude Laurgeau	Professeur, CAOR – Ecole des Mines de Paris	Examinateur
M. François Goulette	Maitre-Assistant, CAOR – Ecole des Mines de Paris	Directeur de thèse

Résumé

Le domaine médical voit apparaître depuis peu des simulateurs pédagogiques, à l'instar de ce qui existe depuis quelques années dans d'autres domaines (aviation, auto-école). En effet, la formation actuelle des chirurgiens se fait à la fois sur des cadavres (ce qui entraîne des difficultés grandissantes et des coûts élevés) et par compagnonnage (avec des chirurgiens expérimentés). Une forte demande est exprimée pour des simulateurs chirurgicaux pédagogiques, surtout dans le domaine de la chirurgie mini-invasive, avec une prise en compte des gestes et des sensations tactiles.

Ce travail de thèse, réalisé au Centre de Robotique (CAOR) de l'Ecole des Mines de Paris, couvre deux problématiques importantes pour la réalisation d'un simulateur chirurgical pédagogique : la modélisation temps réel des tissus mous et la réalisation d'un système haptique dédié à la chirurgie virtuelle.

Nous nous intéressons à la modélisation temps réel des organes déformables, nous mettons en œuvre une méthode physique, qui offre la possibilité de paramétrer le modèle à partir de mesures expérimentales, et de construire une solution générique applicable à une grande variété de matériaux. Il s'agit d'une approche issue de la Mécanique des Milieux Continus ; c'est une synthèse entre deux présentations répandues : les masses-ressorts et les masses tenseurs. Elle se base sur le modèle des matériaux de Saint Venant Kirchhoff qui correspondent à une classe de matériaux hyperélastiques. Une fois modélisé, l'organe virtuel doit interagir avec l'environnement, il s'agit de la détection des collisions (avec les outils) ou auto-collisions (l'organe se plie sur lui-même) ainsi que la gestion de ces collisions. Nous traitons aussi la découpe que peut subir un organe pendant l'opération.

Nous présenterons aussi nos travaux de conception et de réalisation du système de retour d'effort dédié à la chirurgie, le « trocart actif ». Notre système présente un avantage majeur sur l'existant, il peut être utilisé pour une grande variété d'opérations chirurgicales endoscopiques. Il s'agit de l'instrumentation du trocart, outil indispensable pour la chirurgie mini-invasive afin d'assurer au maximum que les outils de chirurgie ne heurtent pas la peau à l'endroit de l'incision. Nous l'appelons « trocart actif ». Nous avons optimisé la conception de ce trocart afin de permettre l'utilisation de plusieurs trocarts actifs dans la même opération de chirurgie virtuelle et évidemment afin d'assurer le bon fonctionnement d'un système haptique pour garantir un bon rendement tactile. Avec ce système nous n'imposons pas le choix de l'outil chirurgical au jeune chirurgien. Celui-ci aura le choix de l'outil chirurgical au cours de l'opération.

Abstract

The medical field is emerging recently some pedagogic simulators like what it exists from several years in other field like flight simulator or car driving simulator. Indeed, the current training of surgeons is performed on corpses (difficult, expensive and not practical) and companionship (with experienced surgeons). These difficulties involve a growing demand for surgical pedagogic simulators, especially in the field of mini-invasive surgery, with a consideration of gestures and tactile sensations.

This thesis work, conducted in the Robotics Center(CAOR) of the Ecole des Mines de Paris, covers two important issues for achieving a surgical simulator : the real time soft tissue modeling and the realization of a haptic system dedicated to the virtual surgery.

In order to model the deformable organ in real time, we conceive a physical method that offers the opportunity to parameterize our model starting from experimental measurements and to build a generic solution that can be applied to a wide variety of materials. Our approach consists of a synthesis between two kinds of presentation used in the field of continuum mechanics : the mass spring and the mass tensor. Our method is based on the Saint Venant Kirchhoff model which corresponds to a class of hyper-elastic materials. Once modelled, the body in ready to interact with the virtual environment. We, thus, propose some mechanisms for collisions detection with surgical tools and with the structure of the body itself. We propose also some mechanisms for collision management. A particular attention is given to the cutting mechanism that the body usually undergoes during surgery.

In the second section, we present our design and realization of a force feedback system dedicated to the virtual surgery. It is a new design of the trocar, an essential tool for the mini-invasive surgery. We call it "active trocar". It is a 5 DOF force feedback system. Our system has a major advantage on what exists. It can be used for a large variety of endoscopic surgical operations because we it is based on the trocar and not on the endoscopic surgery tool. We optimized the design of this trocar in order to allow the use of several devices in the same virtual operation and evidently in order to guarantee good tactile effects. With this system, the surgeon is free to choice the suitable surgical tool and fit into in the trocar.

Remerciements

Les travaux de thèse présentés dans ce mémoire ont été réalisés au sein du Centre de Robotique de l'Ecole des Mines de Paris.

Je commencerais par remercier tous les membres de mon jury de thèse pour l'enthousiasme qu'ils ont porté a mes travaux. J'adresse ma reconnaissance tout particulièrement à Monsieur Claude Laurgeau, directeur du Centre de Robotique de l'École de Mines de Paris, de m'avoir offert l'opportunité de mener à bien cette thèse dans les meilleures conditions et d'avoir présidé mon jury de thèse. Tous mes remerciements à Monsieur François Goulette, pour avoir été mon directeur de thèse, pour son enthousiasme, pour sa disponibilité, pour son avis éclairé pour la rédaction ce mémoire.

Je remercie Monsieur Serge Couvet, de la société Thales, pour les échanges passionnants autour du système de retour d'effort et divers technologies.

Je tiens aussi à remercier tout le personnel du Centre de Robotique (CAOR) de l'École des Mines de Paris dont l'expérience m'a permis d'adopter une démarche scientifique et objective. Mes remerciements à Fawzi Nashashibi, Philippe Fuchs, Brigitte d'Andrea Novel pour leur sympathie et leurs nombreux conseils en particulier lors des Doctorades. Je remercie Joël Senpauroca pour la lecture de cette thèse, son enrichissement du travail sur l'interface haptique et ses nombreux conseils. Mes remerciements vont aussi aux poissons Domitile, Panos, Xavier, Laure. Mes remerciements à Bogdan, Clément, Ayoub, Jorge, Bruno, Sébastien, Konaly, mon coach Christine, David, Wajih, Saber, Mehdi, Antoine, Myriam, Hisham, Ghada, Maher pour leur bonne humeur.

Je n'oublierais pas tous le personnel de l'école des Mines, de l'accueil, de la cafétéria, de la bibliothèque, du service reprographie, Armines et tous ceux avec qui j'ai eu des discussions enrichissantes. Je remercie Thérèse, Joceline, et Elisabeth.

Je tiens à remercier Samer qui m'a accompagné depuis mes études d'ingénieur jusqu'à la fin de ma thèse, Youmi, Rami, Talal, Imad, Nazih et Salah.

Je n'oublierais pas Nancy pour m'avoir soutenu et encouragé, mes beaux parents, ainsi que mes parents pour m'avoir permis et incité à entreprendre des études enrichissantes et passionnantes en espérant que ça soit une science dont les gens tirent profit.

Table des matières

Résumé **1**

Abstract **3**

Remerciements **5**

Introduction **15**

1 Les simulateurs chirurgicaux **19**
1.1 La chirurgie laparoscopique - besoins en simulation 20
1.2 Classification des simulateurs médicaux 21
1.3 Principes techniques d'un simulateur chirurgical 22
1.4 Simulateurs existants 23
1.5 Problématiques 25

2 Modélisation temps réel des organes **27**
2.1 La modélisation des corps déformables 28
2.1.1 Critère pour un bon modèle déformable 28
2.1.2 Modèles non-physiques 29
2.1.3 Modèle déformable physique 29
2.2 Principaux modèles déformables 30
2.2.1 Le modèle masse ressort 30
2.2.2 Le modèle éléments finis 31
2.2.3 Le modèle masse tenseur 32
2.2.4 Le modèle à barres 33
2.2.5 Comparaison des différents modèles 33
2.3 Le modèle masse-ressort-amortisseur détaillé 33
2.3.1 Maillage 35
2.3.2 Homogénéité des masses 36
2.3.3 Identification des raideurs 37
2.3.4 Test rhéologique et essai sur le modèle numérique 38
2.3.5 Limitations du modeles 40
2.4 Intégration numérique 40
2.4.1 Intégration explicite 41

2.4.2 Intégration implicite 42
2.4.3 La méthode d'Euler retrograde 43
2.4.4 Résolution par la méthode du point fixe (Point fixe de Banach) . . 44
2.5 Conclusion 44

3 Modèle viscoélastique non-linéaire de déformation 45
3.1 Introduction 46
3.2 Rappels de la Mécanique des Milieux Continus 46
3.2.1 Rappel des tenseurs 47
3.2.2 Rappel de la loi de Hooke 47
3.2.3 Elasticité non linéaire : L'hyperélasticité 48
3.3 La méthode Masse liaison étendue 49
3.3.1 Configurations initiale et déformée 50
3.3.2 Gradient de déformation, Tenseurs de dilatation et de déformation 54
3.4 Algorithme et implémentation 58
3.5 Validation de la méthode 58
3.5.1 Résultat analytique 59
3.5.2 Comparaison analytique-numérique 60
3.6 Le modèle Visco-élastique 61
3.6.1 Différents modèles de Viscoélasticité 61
3.6.2 Formulation Viscoélastique 62
3.7 Conclusion 64

4 Interaction avec les organes modélisés 67
4.1 Introduction 68
4.2 Détection des collisions 68
4.2.1 Méthodes de détection de collision approximative 70
4.2.2 Méthode choisie 75
4.3 Gestion des collisions 77
4.3.1 Méthodes classiques de gestion des collisions 77
4.3.2 Implémentations testées pour la gestion des collisions 79
4.3.3 Temps réel et stabilité 82
4.4 La découpe 84
4.4.1 Les trois méthodes pour la découpe 84
4.4.2 Les méthodes implémentées 86
4.5 Conclusion 97

5 Système haptique dédié à la chirurgie virtuelle 99
5.1 Les interfaces haptiques 100
5.2 Les interfaces haptiques pour l'apprentissage du geste chirurgical 101
5.3 Conception d'une nouvelle interface : trocart actif 104
5.3.1 Les gestes du chirurgien - analyse du besoin 104
5.3.2 Analyse fonctionnelle d'une nouvelle interface 106
5.3.3 Contraintes techniques et réalisation du cahier des charges 107

5.4 Réalisation du dispositif 110
5.4.1 L'architecture mécanique 110
5.4.2 Choix des composants 112
5.4.3 Architecture électronique 117
5.4.4 Commande du trocart actif 118
5.4.5 Validation du prototype 122
5.5 Conclusion 124

Conclusion Générale et Perspectives **127**

Bibliographie **129**

Table des figures

1.1 Chirurgie abdominale en laparoscopie *United States Surgical Corporation © 1997* . . . 20
1.2 Les trois générations de simulateurs proposées par Satava [Sat96] . . . 22
1.3 Principe d'un simulateur chirurgical . . . 23

2.1 Spline Cubique . . . 29
2.2 Modèle masse-ressort-amortisseur . . . 34
2.3 Structure du fichier de maillage utilisé . . . 35
2.4 Structure de données . . . 36
2.5 Gravité appliquée au cube sans homogénéité de masse volumique . . . 36
2.6 Gravité appliquée au cube avec homogénéité de masse volumique . . . 37
2.7 Différents types de maillage . . . 39
2.8 Force-displacement curves (a) :irregular cube ; (b) :cube 1 ; (c) :cube 5x5x5x ; (d) :cube 10x10x10 . . . 39
2.9 Méthodes d'intégration explicites . . . 41
2.10 Boucle d'intégration . . . 42

3.1 configurations initiale et déformée . . . 50
3.2 test de validation . . . 58
3.3 Comparaison courbes contrainte-déformation analytique et numérique (cube unitaire) . . . 61
3.4 Comparaison courbes contrainte-déformation analytique et numérique . . . 62
3.5 Comparaison courbes contrainte-déformation analytique et numérique en cas de traction . . . 63
3.6 Le modèle de Maxwell . . . 63
3.7 Le modèle de Kelvin-Voit . . . 63
3.8 Comparaison courbes contrainte-déformation analytique et numérique en comportement viscoélastique . . . 64
3.9 Erreur entre les courbes contrainte-déformation analytique et numérique en comportement viscoélastique suite à un grand coefficient d'amortissement 65
3.10 Courbes contrainte-déformation analytique et numérique pour plusieurs tailles de maillage . . . 66

4.1 *Détection de collision point-objet* . . . 69

4.2 *octree* 73
4.3 Détection des collisions 76
4.4 Methode des pénalité 79
4.5 Methode des pénalité 80
4.6 Détection des collisions 81
4.7 Détection des collisions 81
4.8 *Retrait d'un tétraèdre lors de la collision avec l'outil coupant* 84
4.9 *Subdivisions minimisant le nombre de tétraèdres créés [Mor01]* 85
4.10 *Déplacement des sommets pour coïncider avec le chemin du scalpel* 86
4.11 *La notion de proxy* 87
4.12 *La découpe par retrait de tétraèdre* 88
4.13 *algorithme géomètrique de l'ouverture d'un maillage* 89
4.14 *ouverture minimale d'un maillage* 89
4.15 *Projection des sommets le long du chemin du scalpel* 90
4.16 *Gestion des sommets en profondeur* 91
4.17 *Apparition de dégénérescences lors de la phase de déplacement des sommets.* 92
4.18 *Différents tétraèdres singulier de type I ou II [For03]* 93
4.19 *Triangulation de Delaunay* 95
4.20 *Approche par écrasement* 96
4.21 *Comparaison des répartitions des critères en fonction du volume* 98

5.1 Le Phantom Desktop de SensAble Technologies et le Virtuose 6D de Haption 102
5.2 L'Omega.3 de la société ForceDimension 102
5.3 A gauche : le Virtual Laparoscopic interface. Au centre : le Laparoscopic Impulse Engine. A droite : Le Laparoscopic Surgical Workstation 103
5.4 A gauche : Le HIT Force Feedback Device. Au centre : le simulateur Vest Systeme One. A droite : Le simulateur LapMentor 104
5.5 Les trocarts et Les outis de chirurgie endoscopique 105
5.6 Opération d'endoscopie : Les trocarts placés places sur le corps humain.(http ://www.maitrise-orthop.com) 105
5.7 Les 5 degrés de liberté du trocart actif 107
5.8 Coeur et berceau du trocart actif 108
5.9 Schéma des différentes parties du simulateur 109
5.10 Habillage de l'outil pour la translation 111
5.11 Principe de translation 113
5.12 Le module de la rotation propre 114
5.13 Le Berceau : tangage et roulis 115
5.14 Dispositif du tangage 115
5.15 Dispositif du roulis 116
5.16 Architecture électronique de l'interface 117
5.17 Carte de commande fournie 118
5.18 Nouvelle architecture électronique de l'interface 119
5.19 Schéma et notations 120
5.20 Le modèle CAO du système 125

5.21 Le prototype réalisé 126

Introduction

Le domaine médical voit apparaître depuis peu des simulateurs pédagogiques, à l'instar de ce qui existe depuis quelques années dans d'autres domaines (aviation, auto-école) : opérations d'ophtalmologie, échographie virtuelle, simulation d'amniosynthèse et surtout simulation de chirurgie mini-invasive. Les conditions dans lesquelles le chirurgien opère pendant une intervention de chirurgie mini-invasive sont modifiées par rapport à une intervention classique. Par exemple, les mouvements du chirurgien sont contraints, car les instruments se déplacent uniquement par un point fixe, le trocart. Il y a aussi une désynchronisation entre la vue et les gestes du chirurgien qui visualise l'opération sur un écran et n'a plus une vision directe sur le corps du patient. Il perd également la sensation de profondeur.

Le chirurgien doit donc apprendre à maîtriser cette technique afin d'acquérir la dextérité requise pour la manipulation des instruments et s'habituer à une nouvelle coordination entre les yeux et les mains.

La formation actuelle des chirurgiens se fait à la fois sur des cadavres (ce qui entraîne des difficultés grandissantes et des coûts élevés) et par compagnonnage (avec des chirurgiens expérimentés). Une forte demande est exprimée donc pour des simulateurs chirurgicaux pédagogiques, surtout dans le domaine de la chirurgie mini-invasive, avec une prise en compte des gestes et des sensations tactiles.

Buts du travail

L'objectif de mon travail porte sur la conception et le développement d'un simulateur de chirurgie mini-invasive. Plus particulièrement, les objectifs sont :

- Implémentation d'un modèle physique pour les organes déformables. Dans ce contexte nous allons faire une étude comparative des différents modèles physiques
- Décrire quelques techniques pour la détections de collision entre les organes et les outils de chirurgie ainsi que les collisions des organes.
- Aborder les changements de topologie, en particulier la découpe des tissus mous.
- Concevoir et réaliser un système de retour d'effort dédié à la chirurgie pour donner la sensation de toucher les organes modélisés.

Contexte du travail

Le travail de ma thèse s'inscrit dans le cadre du projet **SIMV@L** répondant à l'appel du Réseau National des Technologies pour la Santé (**RNTS**). Il s'agit de concevoir et réaliser un simulateur générique destiné à la formation et à la certification en chirurgie urologique. Ce projet est constitué de huit partenaires sous la coordination d'un industriel spécialiste dans la simulation.

Pour le travail sur les corps déformables, nous avons collaboré avec le Centre de Mise en Forme des Matériaux (**CEMEF**) de l'Ecole des Mines de Paris à Sophia-Antipolis, dans le cadre de la thèse d'Audrey Paccini sur le développement d'un modèle éléments finis 3D appliqué à la simulation d'opérations chirurgicales des tissus mous [Pac05].

Pendant ce travail, j'ai suivi et co-encadré deux stages de DEA. Avec Xavier Brun nous avons travaillé sur la découpe des corps déformables[Br04]. Avec Antoine Arnail nous avons travaillé sur la détection des collisions, auto-collisions et la gestion des collisions[Arn05].

Nous avons inscrit la réalisation de l'interface haptique de chirurgie dans le cadre d'un système pédagogique de mécatronique à l'École des Mines de Paris (**ENSMP**). Pour la réalisation de ce projet, deux équipe de 14 élèves chacune ont été constituées : 10 élèves faisant partie de l'ENSMP (élèves de 2ième année) et 4 élèves du Lycée Louis Armand de Nogent(STS CPI).

Contributions

Les contributions de ce travail sont :

- Un exposé des problématiques et contraintes qui pèsent sur la mise en oeuvre d'un simulateur complet de chirurgie mini-invasive.
- un exposé des modèles de corps déformables existants ainsi que des améliorations du modèles masse-ressort[CGC04][CGC04+].
- Un modèle de corps déformable, qui est une approche issue de la Mécanique des Milieux Continus. Ce modèle représente un avantage de temps de calculs pour la simulations temps réel ainsi qu'une représentation fidèle de la loi de comportement de l'organe. Nous révélons cet avantage suite à une étude comparative concernant les différents modèles physiques[GC06].
- Analyse et implémentations des méthodes de détection des collisions, de gestion des collisions et de découpe de l'organe déformable[BCG05].
- Conception et réalisation d'une interface de retour d'effort pour la chirurgie mini-invasive [CG08].

Plan de la thèse

Dans le chapitre 1, nous introduisons d'une manière générale les simulateurs chirurgicaux ainsi que leurs principes techniques puis nous présentons les principaux simulateurs chirurgicaux existants. Nous parlons des trois contraintes majeures pour des bons simulateurs chirurgicaux : le modèle déformable, l'interaction avec ce modèle et le système de retour d'effort.

Dans le chapitre 2, nous définissons les critères pour un bon modèle déformable, puis nous présentons une étude comparative concernant les différents modèles physiques. Puis en se basant sur le modèle masse-ressort nous introduisons le maillage et l'intégration numérique utilisés pour les corps déformables.

Dans le chapitre 3, nous proposons notre approche pour la simulation temps-réel des corps déformables. Cette approche est une méthode physique, qui offre la possibilité de paramétrer le modèle à partir de mesures expérimentales, et de construire une solution générique applicable à une grande variété de matériaux. Elle repose sur un modèle continu de la matière, ce qui la rend indépendante de la topologie du maillage et de la forme de l'objet. Elle permet aussi de simuler des changements de topologie, indispensables pour la chirurgie virtuelle. Elle offre en outre une grande flexibilité de mise en œuvre. Il s'agit d'une approche issue de la Mécanique des Milieux Continus pour modéliser un comportement viscoélastique non linéaire, appliquée aux matériaux de Saint Venant Kirchhoff qui correspondent à une classe de matériaux hyperélastiques.

Dans le chapitre 4, nous traitons l'interaction avec/entre les organes déformables (Détection des collisions, auto-collisions, gestion des collisions, découpe des organes). L'utilisateur manipule les organes déformables par une Interface Homme Machine (IHM) dédiée. Le système doit être capable de savoir quand est-ce que l'outil virtuel entre en contact (détection des collisions) avec l'organe déformable et aussi comment cet organe se comporte lors d'un tel contact. Ajoutons qu'un organe peut entrer en contact avec un autre organe ou avec lui-même (auto-collisions). L'organe peut être découpé lors d'une opération chirurgicale.

Dans le chapitre 5, nous présentons nos travaux de conception et de réalisation du système à retour d'effort dédié à la chirurgie. Nous réalisons une interface qui imite un trocart, outil utilisé en chirurgie mini-invasive comme dispositif pour entrer les instruments de chirurgie afin d'assurer au maximum que ces instruments ne heurtent pas la peau du patient à l'endroit de l'incision. Nous l'appelons « trocart actif ». Il s'agit d'un système de retour d'effort à 5 degrés de liberté DDL.

Chapitre 1

Les simulateurs chirurgicaux

Dans ce chapitre nous introduisons d'une manière générale les simulateurs chirurgicaux en évoquant leurs principes techniques et les problématiques qui sont au coeur des travaux de recherche pour la conception et la réalisation d'un bon simulateur chirurgical.

1.1 La chirurgie laparoscopique - besoins en simulation

Les progrès réalisés ces dernières années en chirurgie ont été marqués par un nouveau type de chirurgie dite chirurgie non invasive. En effet, la tendance générale en chirurgie est de réduire les traumatismes des opérations, en diminuant notamment la taille de l'accès de la zone à opérer. En chirurgie endoscopique, le principe revient à ne pas ouvrir le corps du patient comme on avait l'habitude de le faire, mais à pratiquer des micro incisions qui serviront de passage pour l'insertion d'outils chirurgicaux (pinces, ciseaux, aspirateurs, pointes coagulantes, etc.).

Le premier constat est que la vision directe de la zone d'intervention du chirurgien est supprimée. Par conséquent, il est nécessaire d'introduire une micro caméra par l'un des trocarts : c'est ce qu'on appelle l'endoscope. Cet endoscope est raccordé par un faisceau de fibres optiques ou via une caméra CCD à un moniteur couleur et on visualise l'image prise par la caméra. On visualise ainsi l'intérieur du corps du patient : le chirurgien opère donc via un écran vidéo (figure 1.1). C'est une technique qu'il faut maîtriser, de nou-

Fig. 1.1 – Chirurgie abdominale en laparoscopie *United States Surgical Corporation © 1997*

veaux repères, de nouvelles sensations qu'il faut acquérir. Pour cela il faut s'entraîner, et

le malade n'est certainement pas le cobaye idéal pour faire ses premiers pas. La plupart des chirurgiens se sont entraînés sur des animaux, en particulier des cochons. Si on fait abstraction de toutes questions éthiques, les animaux ont été un bon moyen d'apprentissage, mais à l'heure actuelle, il est interdit d'opérer des animaux dans nombre de pays, et le coût est élevé. Il est encore possible d'opérer sur des cadavres humains, mais on perd énormément de réalisme du fait de l'absence de saignement, de respiration et de battement cardiaque, et leur coût reste élevé. Les médecins peuvent s'entraîner sur des systèmes mécaniques prévus à cet effet, nommés « endotrainers ». Les endotrainers permettent au chirurgien d'utiliser les mêmes types d'instruments que ceux qui sont utilisés en chirurgie mini-invasive, insérés par les ouvertures d'une boîte transparente. Le chirurgien peut ensuite interagir avec des objets en plastique représentant les divers organes. Ceci constitue une approximation très grossière mais surtout statique de la réalité, qui ne restitue pas la nature dynamique des tissus, la déchirure potentielle des organes et autres facteurs.

Les chirurgiens n'ont donc plus de choix : en l'absence d'autres méthodes d'entraînement réalistes, ils font leurs premiers essais sur le malade, c'est le cas des méthodes d'apprentissage par compagnonnage. Les risques d'erreurs sont donc beaucoup plus importants et le danger d'une médecine à deux vitesses n'est pas à écarter : les chirurgiens expérimentés seront beaucoup plus chers que les novices. Il est donc indispensable que les chirurgiens aient rapidement une bonne maîtrise de leurs instruments et des gestes opératoires avant d'opérer leur patient. Il en va de la santé du malade et de la crédibilité de la chirurgie. C'est pourquoi on a recours à la simulation, surtout que les techniques opératoires de la chirurgie mini-invasive ne nécessitent pas de visualisation directe du champ opératoire. Les simulateurs chirurgicaux restent un environnement sûr et confortable pour l'apprentissage. Ces simulateurs permettent, en plus de l'apprentissage des gestes de manipulation des instruments chirurgicaux, de provoquer des événements inattendus à n'importe quel moment, l'élève chirurgien pourra donc être confronté à des difficultés qui ne sont rencontrées que très rarement lors des opérations réelles. Un simulateur chirurgical peut même servir à un chirurgien expérimenté, ce dernier pourra se préparer en amont d'une opération chirurgicale délicate pour avoir la main lors de l'intervention directe avec le patient. Enfin, un simulateur chirurgical sert à standardiser l'évaluation en chirurgie à partir de critères précis.

1.2 Classification des simulateurs médicaux

Le professeur R. Satava a proposé une classification des différentes générations de simulateurs médicaux [Sat96].

La première génération de simulateur considère uniquement l'aspect géométrique de l'anatomie et permet uniquement la " navigation " interactive à l'intérieur du corps humain. Cette interaction reste assez limitée.

La seconde génération prend en compte la réalité physique des tissus et en particulier leur caractère déformable. De tels simulateurs peuvent représenter l'interaction d'instruments chirurgicaux avec les organes, notamment les procédures de déformation et de

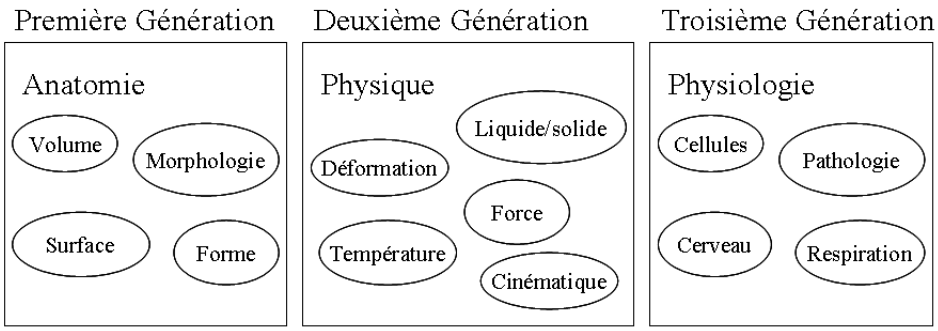

FIG. 1.2 – Les trois générations de simulateurs proposées par Satava [Sat96]

découpe.

La troisième génération de simulateurs médicaux prend en compte l'impact de l'intervention chirurgicale sur la physiologie du corps humain. Ainsi, la découpe d'un vaisseau sanguin aura un effet sur la pression sanguine du patient ce qui va perturber le fonctionnement d'autres organes.

1.3 Principes techniques d'un simulateur chirurgical

La simulation chirurgicale vise, à travers divers moyens informatiques et mécaniques, à reproduire les sensations visuelles et gestuelles d'un chirurgien lors d'une opération chirurgicale.

La figure 1.3 montre le principe technique d'un simulateur chirurgical. Il s'agit en général de trois unités : la première est l'unité d'affichage. La deuxième unité est le modèle déformable représentant la géométrie et le comportement biomécanique de l'organe à simuler. Cette unité est le centre d'intérêt dans un simulateur. La troisième unité est le système d'interaction avec le corps déformable. Cette interaction se fera en général par l'intermédiaire d'une interface mécanique ressemblant au matériel réel utilisé dans une opération réelle. Cette interface est parfois équipée d'un dispositif à retour d'effort.

La visualisation se fait sur un écran d'ordinateur. Les sensations tactiles sont assurées par l'utilisation de systèmes à retour d'effort nommés systèmes haptiques. Ces systèmes à retour d'effort ont été introduits dans des systèmes de télé-opération visant à transmettre à un robot esclave les actions effectuées sur un robot maître [Gre95]. L'utilisation du système haptique en télé-opération représente un avantage de 43% par rapport aux scénarios sans retour d'effort [Hop02]. Une autre étude menée à l'université de Harvard [Wag02], montre que l'absence d'un système à retour d'effort peut augmenter d'au moins 50% la force appliquée sur un tissu pendant une opération chirurgicale. L'utilisation d'un système à retour d'effort est un point fort pour un simulateur de chirurgie, il est démontré qu'un système haptique contribue positivement à l'évaluation d'un simulateur chirurgical [Mar96].

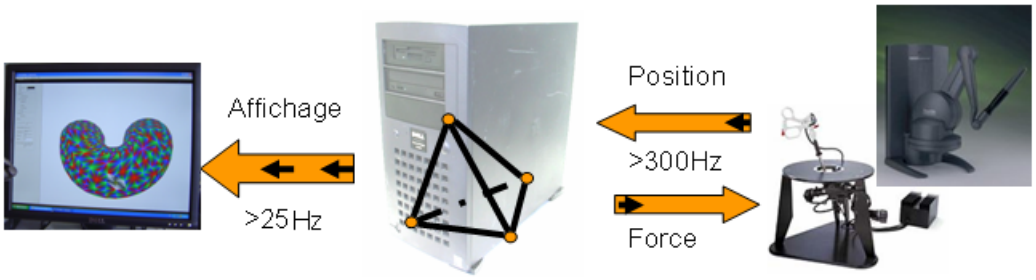

Fig. 1.3 – Principe d'un simulateur chirurgical

L'unité centrale modélisant le corps déformable doit fournir l'affichage après le traitement de la déformation et assurer en même temps l'interaction avec le dispositif mécanique. Il s'agit alors de deux boucles principales qui sont asynchrones. La boucle gérant le dispositif de 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 300-1000Hz [Shi95]. La seconde boucle sert à assurer une vision de la déformation et de l'interaction. Cette boucle met à jour l'affichage de la scène à une fréquence de 25Hz. Il existe une troisième boucle interne au modèle déformable qui détecte la collision entre l'instrument et le corps déformable et gère le comportement du corps modélisé, c'est à dire traduit les collisions en contraintes qui sont appliquées au modèle déformable afin de modéliser le contact, et calcule la déformation et la force exercée sur l'instrument. Cette force est renvoyée au système à retour d'effort.

1.4 Simulateurs existants

La simulation est couramment utilisée dans le domaine de l'aviation commerciale puisqu'elle est la méthode la plus rentable pour former les pilotes. Les pilotes doivent atteindre un certain degré d'adresse dans le simulateur avant d'être autorisés à voler aux commandes d'un avion particulier et doivent passer régulièrement des épreuves de compétence dans le simulateur pour conserver leur brevet. Les Forces armées utilisent des techniques semblables pour favoriser l'acquisition des habiletés de base. Elles considèrent également que la simulation est utile dans la formation visant l'acquisition des habiletés de combat dans des situations tactiques complexes. La collectivité médicale commence à utiliser la simulation dans quelques domaines pour former certaines catégories de personnel médical telles que les chirurgiens et les anesthésistes.

De nombreux simulateurs chirurgicaux sont déjà mis sur le marché par un certain nombre de sociétés, et sont toujours des sujets de recherche et de discussion dans des grands centres de recherche.

Le premier simulateur de chirurgie laparoscopique pourrait être le Minimally Invasive Surgery Trainer (MIST), développé dès 1995 et commercialisé en 1997 par la société Virtual Presence Ltd. C'est un simulateur simple qui consiste surtout à reproduire l'in-

terface habituelle du médecin et lui permettre d'apprendre à manipuler et à connaître les organes représentés. Ce simulateur est caractérisé par son environnement pédagogique permettant aux étudiants de bien se familiariser avec les instruments de la chirurgie laparoscopique. Ce simulateur a bénéficié de nombreuses améliorations après l'acquisition des activités médicales de Virtual Presence Ltd par la société Mentice[1].

Après Virtual Presence Ltd, un grand nombre d'entreprises se sont lancées dans la simulation de chirurgie. C'est le cas de la société Boston Dynamics, dont les simulateurs de chirurgie avec retour d'effort permettent la simulation d'arthroscopie.

Certains simulateurs proposent des modèles déformables, généralement surfaciques. Le modèle déformable le plus utilisé étant le modèle masse-ressort, du fait de la simplicité de sa mise en œuvre. L'opération la plus simulée est la cholécystectomie c'est-à-dire l'ablation de la vésicule biliaire, disponible sur le LapChole de la société Xitact[2], le LapSim de Surgical Science[3] ou encore le RLT de ReachIn[4].

Le produit Virtual Patient de Xitact permet de simuler des gestes de chirurgie mini-invasive avec retour d'effort. Ce simulateur est issu des recherches menées par l'équipe des professeurs Hannes Bleuler et Reymond Clavel de l'Institut de Systèmes Robotiques[5] de Lausanne (EPFL).

Le RLT de ReachIn est un simulateur de chirurgie laparoscopique avec retour d'effort, et peut être utilisé pour un grand nombre d'applications.

On trouve aussi des simulateurs d'opérations gynécologiques ou d'appendicectomie comme par exemple le LapMentor de la société Simbionix[6] ou le VSOne de Select-It[7].

La société Simbionix commercialise aujourd'hui quatre simulateurs, le GI Mentor pour l'endoscopie, le PERC Mentor pour la radiographie et la fluographie, le URO Mentor pour l'urologie, et le Lap Mentor qui est un simulateur multidisciplinaire pour un entraînement à la chirurgie laparoscopique.

Le VSOne de Select-It a été réalisé dans le cadre de KISMET à l'université de Karlsruhe. Il a été conçu comme un prototype de chirurgie laparoscopique abdominal [Kuh96] [Küh97]. Il a été ensuite étendu à la chirurgie gynécologique avec les travaux de thèse de H.K. Çakmak [Çak00][Küh99]. Il est caractérisé par une interaction complète avec les modèles déformables. Il utilise un système de retour d'effort développé dans la même université.

De manière générale, les produits commercialisés de nos jours sont basés sur des modèles relativement simplifiés, afin de garantir une bonne fluidité de la visualisation et du retour d'effort. Le laboratoire d'Informatique Fondamentale de Lille (LIFL) a développé un simulateur de cœlioscopie gynécologique SPIC muni d'une interface sans retour d'effort[Mes00].

[1] http ://www.mentice.com
[2] http ://www.xitact.com
[3] http ://www.surgical-science.com
[4] http ://www.reachin.se
[5] http ://www.epfl.ch/isr/
[6] http ://www.simbionix.com
[7] http ://www.select-it.com

1.5 Problématiques

Il existe plusieurs contraintes pour la réalisation d'un bon simulateur. Ces problématiques sont surtout la modélisation temps réel des corps déformables basée sur la physique, l'affichage réaliste, la détection des collisions et auto-collisions, le contrôle de dispositifs à retour d'effort ainsi que le découplage de la simulation avec ces dispositifs, sans oublier l'affichage réaliste de structures et les effets qui pourront immerger l'utilisateur dans l'environnement virtuel. Nous détaillons ces contraintes ci-dessous.

La modélisation de corps déformable basée sur la physique : Les corps déformables sont le noyau d'un simulateur chirurgical ainsi que l'interaction avec ces corps, c'est-à-dire la saisie et la découpe. C'est le modèle déformable qui crée un compromis entre le réalisme et l'interactivité. En d'autres termes, c'est soit un réalisme comportemental, soit un réalisme d'interaction et de temps de réponse. C'est le problème du temps réel, fortement rencontré quand il s'agit d'un simulateur avec retour d'effort. Le choix du modèle déformable doit être tiré des caractéristiques biomécaniques de l'organe réel à simuler.

L'interaction avec/entre les organes déformables : L'utilisateur manipule les organes déformables par une IHM dédiée. Le système doit être capable de savoir quand l'outil virtuel entrera en contact (détection des collisions) avec l'organe déformable et aussi comment cet organe se comportera lors d'un tel contact. Ajoutons qu'un organe peut entrer en contact avec un autre organe ou avec lui-même (auto-collisions). L'organe peut être découpé lors d'une opération chirurgicale. Cette possibilité de découper des tissus est d'une très grande importance pour tout simulateur chirurgical. Cette contrainte influe sur le choix de la modélisation des tissus, car la topologie de l'organe virtuel doit pouvoir être changée au cours du temps.

L'Interface Homme-Machine dédiée à la chirurgie IHM : C'est un outil indispensable pour la création d'environnement virtuel de chirurgie mini-invasive. Afin d'immerger l'utilisateur dans l'environnement d'une opération chirurgicale, cette interface est souvent un dispositif à retour d'effort. Ce dispositif a pour fonction d'une part de permettre de guider un outil virtuel dans un environnement virtuel, et d'autre part de transmettre en permanence des forces à l'utilisateur afin de lui assurer la sensation du toucher.

A ces éléments peuvent s'ajouter des effets spéciaux pour l'affichage des organes pour donner l'illusion d'une immersion totale dans l'environnement d'une vraie opération chirurgicale. Il est nécessaire de modéliser correctement les textures des tissus, les effets d'éclairage spécifiques d'un endoscope et différents effets spéciaux.

Dans tous les cas nous devons respecter la contrainte du **temps réel**. Cette contrainte est omniprésente pour les différentes parties d'un simulateur chirurgical interactif. L'utilisateur doit pouvoir s'immerger dans un environnement virtuel semblable à l'environnement d'une opération réelle. Il doit pouvoir interagir sans latence avec les organes déformables modélisés, les voir se déformer, se déchirer ou se découper et sentir une force de retour lors de cette interaction à travers le système à retour d'effort. Comme nous

l'avons mentionné au paragraphe §1.3, pour la visualisation, les fréquences acceptables sont de l'ordre de 25Hz alors que pour le retour d'effort, elles montent à 300-1000Hz.

Chapitre 2

Modélisation temps réel des organes

L'étude des corps déformables a connu de grands progrès dans des applications d'études des structures métalliques (poutres, ponts,...). Ces calculs sont très exigeants en ressources informatiques mais ne nécessitent pas une solution temps réel. Ensuite on s'est intéressé à la modélisation des corps déformables pour d'autres applications telles que les jeux vidéo, les animations 3D et la simulation des tissus pour la couture. Nous nous intéressons aux déformables qui sont aujourd'hui le noyau des simulateurs chirurgicaux.

2.1 La modélisation des corps déformables

L'étude des corps déformables a connu de grands progrès dans les applications des études des structures métalliques (poutre, ponts,...). Ces calculs sont très exigeants en ressources informatiques mais ne nécessitent pas une solution temps réel. Ensuite on s'est intéressé à la modélisation des corps déformables pour d'autres applications telles que les jeux vidéo, les animations 3D et la simulation des tissus pour la couture.
Comme nous l'avons présenté dans le chapitre précèdent, la modélisation temps réel des organes déformables est le noyau d'un simulateur chirurgical. Cette modélisation est la principale contrainte pour réaliser un bon simulateur. C'est le modèle déformable qui réalise un compromis entre le réalisme et l'interactivité. En d'autres termes, c'est soit un réalisme comportemental, soit un réalisme d'interaction et de temps de réponse, c'est le problème temps réel, fortement rencontré quand il s'agit d'un simulateur avec le retour d'effort. Dans ce chapitre nous définissons les critères d'un bon modèle déformable, puis une étude comparative concernant les principaux modèles physiques. Nous traitons en particulier le modèle masse-ressort pour finir par le problème de l'intégration numérique qui est une étape commune et nécessaire dans la mise en œuvre de tous les modèles des corps déformables.

2.1.1 Critère pour un bon modèle déformable

Avant de présenter les principaux modèles de tissus mous et pour mieux savoir les avantages et les inconvénients de chaque modèle, il est indispensable de définir les critères qui sont en même temps les contraintes, que nous devrons doter à un organe déformable. Nous résumons ces critères en trois fonctions principales :

La modélisation réaliste des différents matériaux : le modèle déformable doit pouvoir représenter plusieurs types d'organes. Un organe peut avoir soit un comportement élastique, soit viscoélastique, soit hyperélastique... outre le comportement, l'organe peut subir pendant une opération chirurgicale de grandes déformations, donc le modèle virtuel doit pouvoir interagir correctement lors de ces déformations, c'est surtout le cas pour la préhension d'un organe.

La découpe : outre la modélisation du comportement lors d'une déformation ou d'une interaction, l'organe subit pendant une opération chirurgicale un changement de structure géométrique. Ce changement de topologie peut avoir un impact sur le comportement de l'organe ainsi que sur le temps de calcul.

L'interactivité : c'est une contrainte majeure pour un bon modèle déformable. Rappelons que nous nous intéressons à la modélisation des tissus mous pour la simulation médicale. Cette application nécessite une interactivité immédiate, d'où l'aspect temps-réel de la modélisation, ce qui n'est pas le cas pour l'étude des déformations des structures métalliques. [Pac05] a travaillé sur la modélisation des organes déformables sans s'intéresser à l'aspect temps réel, ses travaux s'inscrivent dans le cadre de l'étude du comportement des organes humains.

2.1.2 Modèles non-physiques

Jusqu'à la fin des années 1980, les modèles non-physiques étaient les seuls à être utilisés en animation et en simulation. Ces modèles ne possédant aucune justification liée aux lois de la physique, les déformations mécaniques basées sur ces modèles, font intervenir des fonctions cinématiques ou des constructions géométriques.
Notons par exemple les splines, dans cette représentation, la courbe ou la surface est représentée par un ensemble de points de contrôle. La déformation d'un tel objet est due à un déplacement des points de contrôles ou bien à l'ajout ou à la suppression des points de contrôles.

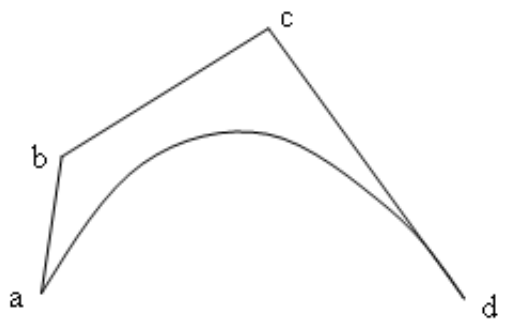

Fig. 2.1 – Spline Cubique

Considérons par exemple la spline cubique de la figure 2.1, cette courbe est définie mathématiquement par le positionnement des points de contrôle. Cette courbe est une interpolation des points a et d, elle est tangente sur les deux côtés du polygone de contrôle. Le déplacement d'un des points de contrôles change la courbe d'une manière prédictible.
Bien que cette méthode soit géométrique pure, il y avait des travaux pour lier les caractéristiques biomécaniques. Dans [FLMC02] les Splines sont utilisées pour modéliser les intestins, la position des points de contrôles forme les degrés de liberté des équations de Lagrange, et les diverses énergies (cinétiques, amortissements, déformations) sont ainsi exprimées en fonction de ces positions et de leurs dérivées.

2.1.3 Modèle déformable physique

L'organe dont on désire simuler le comportement a une structure anatomique définie, c'est à dire un certain volume de tissu biologique que l'on voudra toucher, et qui se déforme selon une loi de comportement physique, issue des lois de la mécanique des milieux continus.
La résolution d'une telle loi par les procédés informatique nécessite une discrétisation du domaine sur lequel on travaille, c'est à ce moment qu'intervient l'approche géométrique du domaine. Il existe plusieurs façons pour représenter discrètement un domaine, soit

par exemple par un ensemble de point, de sphère ou de polygones etc.
La recherche pour la modélisation des objets déformables, surtout des applications médicales est en plein développement. Il existe plusieurs modèles pour présenter les organes déformables. Les principaux modèles sont présentés dans [Nie00] : le modèle masse-ressort-amortisseur [Ber03] [Mon01] que nous détaillons dans la suite, la méthode des éléments frontières [Bou00] qui prend en compte l'objet en surface et évite alors d'avoir à créer un maillage volumique puisque seule la surface doit être maillée. Malheureusement, parce qu'elle exploite abondamment les propriétés de linéarité, il n'est a priori pas possible d'adapter cette méthode à des lois mécaniques plus complexes que l'élasticité linéaire, ni de procéder à des changements de topologie dans l'objet représenté. La méthode des éléments finis [Bro01][Mes02], apparaît comme la plus intéressante pour simuler des déformations mécaniques avec une bonne précision physique. Elle est basée sur un modèle volumique et continu de matière : le système est découpé en un maillage d'éléments volumiques et à l'intérieur de chaque élément les champs de forces et de déplacements sont approximés par des fonctions d'interpolation continues. Elle offre aussi une certaine flexibilité, car à l'aide du choix des fonctions d'interpolation il est possible de modifier la précision des calculs tout en conservant le même maillage. Malheureusement les calculs qu'elle implique sont en général lourds et on a longtemps considéré que cette méthode était inutilisable pour des applications en temps réel. [Cot00] ont changé la manière de résoudre les équations du système d'éléments finis en passant à une résolution locale et itérative, c'est le méthode des masses-tenseurs.

2.2 Principaux modèles déformables

2.2.1 Le modèle masse ressort

Terzopoulos a été le premier à voir l'intérêt de concevoir des modèles de déformation basés sur des lois physiques, universelles, qui puissent être appliqués à un ensemble de situations réelles le plus large possible. Il a réussi à établir le lien entre le monde de la mécanique et celui de l'animation graphique en empruntant les théories et les méthodes de l'ingénierie mécanique pour les appliquer à la simulation d'objets déformables [Ter87].

Les modèles de Terzopoulos ne visaient pas à réaliser des applications en temps réel, mais ont ouvert la voie à toute une série de travaux visant à trouver les algorithmes et les techniques de calcul appropriés à la simulation de déformations en temps réel. Avec l'émergence de besoins de simulation dans le domaine biomédical, un des premiers modèles à s'être répandu a été celui des masses ressorts. Il consiste à discrétiser un objet en un maillage surfacique ou volumique et à attribuer une masse à chaque nœud de ce maillage. Les interactions entre nœuds voisins sont modélisées par des liens élastiques, le plus souvent linéaires, comme si les nœuds étaient reliés entre eux par des ressorts. Il ne s'agit donc pas d'un véritable modèle physique 3D car il ne représente pas la matière sous forme continue, mais plutôt d'une extension en 3D d'un modèle physique unidimensionnel. C'est un modèle physique mais il ne correspond pas à la réalité biomécanique de l'organe.

Le modèle des masses ressorts est bien adapté à la simulation de déformations élastiques en temps réel. Toutefois les résultats obtenus en terme de comportement mécanique sont assez peu réalistes, notamment si l'on se limite à un maillage surfacique. Une solution pour en améliorer la qualité consiste à raffiner le maillage, mais les calculs nécessaires à la résolution du système d'équations différentielles deviennent alors très lourds : il apparaît que pour un niveau de qualité équivalent, cette méthode nécessite un maillage plus fin et des calculs plus longs que la méthode des éléments finis. A cela il faut ajouter que le comportement dépend fortement de la topologie du maillage, autrement dit de la configuration avec laquelle les différents noeuds sont reliés entre eux par des ressorts, configuration qui n'est pas unique pour un ensemble de noeuds donné.

2.2.2 Le modèle éléments finis

La méthode des éléments finis apparaît comme la plus intéressante pour simuler des déformations mécaniques avec une bonne précision physique. Elle est basée sur un modèle volumique et continu de matière : le système est découpé en un maillage d'éléments volumiques et à l'intérieur de chaque élément les champs de forces et de déplacements sont approximés par des fonctions d'interpolation continues. Elle offre aussi une certaine flexibilité, car à l'aide du choix des fonctions d'interpolation il est possible de modifier la précision des calculs tout en conservant le même maillage. Malheureusement les calculs qu'elle implique sont en général lourds et on a longtemps considéré que cette méthode était inutilisable pour des applications en temps réel. Pour des renseignements généraux concernant la mécanique des milieux continus et la méthode des éléments finis, le livre de [Rap98] est particulièrement intéressant.

Bro-Nielsen et Cotin ont démontré le contraire en introduisant un certain nombre d'innovations [Bro98]. Avec un modèle mécanique élastique linéaire, $\sigma = E\varepsilon$ où σ représente le tenseur des contraintes, ε celui des déformations et E le module d'Young, la discrétisation en éléments finis des équations d'équilibre d'un système volumique conduit à un système linéaire d'équations de dimension 3n, n étant le nombre de noeuds du maillage : f = Ku où u est le champ des déplacements dans le système, f est le champ des forces extérieures appliquées au système, K est la matrice de raideur du système. Bro-Nielsen et Cotin ont introduit une condensation de ce système, consistant à extraire les noeuds de surface. En effet, seuls les noeuds de surface jouent un rôle dans la visualisation du résultat ou le retour de forces. La dimension du système à résoudre tombe ainsi à 3 ns, ns étant le nombre de noeuds de la surface du maillage. La matrice est alors inversée explicitement. Cette opération est très longue, mais elle peut être effectuée dans une phase préliminaire de pré-calculs. Puis au cours de la phase de simulation en temps réel, la matrice inversée est utilisée pour le calcul des déplacements. Le produit est rapide à effectuer car le vecteur force contient en général un grand nombre de termes nuls. Le comportement obtenu est quasi-statique, car on a supposé le système à l'équilibre.

Cette méthode qui permet de réaliser des applications en temps réel s'avère inutilisable lorsque la topologie du maillage change. Or, pour des simulateurs chirurgicaux, le maillage doit pouvoir être déchiré, coupé...etc. ce qui engendre une nouvelle matrice de raideur et alors, les pré-calculs effectués ne sont plus utilisables. Récemment, de nouveaux

modèles à éléments finis autorisant les changements topologiques ont été introduits, il s'agit des modèles masses tenseurs.

2.2.3 Le modèle masse tenseur

Cotin et al.[Cot00] se sont attachés à modifier la méthode par éléments finis pour y introduire la possibilité de modifier la topologie . Leur idée a été de changer la manière de résoudre les équations du système d'éléments finis en passant à une résolution locale et itérative. Ils utilisent des matrices de raideur 3x3 pour les sommets appartenant au même tétraèdre. A partir de la discrétisation du champ des déplacements sur un élément fini, on peut exprimer l'énergie de déformation de l'élément en fonction des déplacements de ses sommets, puis dériver cette expression pour obtenir la force élastique exercée sur chacun des sommets. On somme alors toutes les forces qui agissent sur le sommet pour obtenir la résultante en ce point qui sera ensuite intégrée pour obtenir le déplacement du point. Dans son principe, la méthode des masses tenseurs est assez semblable à celle des masses ressorts car dans les deux cas, le système d'équations dynamiques est résolu itérativement et localement. Cependant les deux modèles sont physiquement très différents car les masses ressorts sont une représentation discrète et dépendante de la topologie du maillage, tandis que la méthode des masses tenseurs est basée sur une représentation continue de la matière qui est indépendante de la topologie.

Cet algorithme est malheureusement plus lourd en temps de calcul que l'algorithme quasi-statique global pour éléments finis (section précédente). Pour réduire le temps de calcul, ils ont aussi proposé de combiner les deux méthodes dans un seul système hybride [Cot00]. Puisque les structures où se produisent les changements de topologie ne représentent en général qu'une petite portion de l'organe modélisé, l'algorithme des masses-tenseurs est confiné à cette portion. Les structures environnantes qui ne se déforment que faiblement mais contribuent au réalisme de la simulation sont quant à elles décrites par l'algorithme quasi-statique global. Comme les deux algorithmes sont basés sur la même modélisation physique, ce découpage ne modifie pas le comportement du système, l'ensemble se comporte de la même façon qu'un système élastique linéaire unique.

Deux autres équipes, Nienhuys - van der Stappen [Nie00] et Laugier - Mendoza [Lau03], ont réussi à utiliser la méthode entièrement masse tenseur pour simuler la découpe de tissus mous. L'énergie de déformation est obtenue à l'aide du tenseur de déformation de Green-Lagrange simplifié pour Nienhuys, ce qui n'autorise que les petits déplacements, et générale pour Mendoza, d'un matériau isotrope et élastiquement linéaire appliqué au tétraèdre correspondant. La force subie par un sommet est finalement la somme des forces générées par les tétraèdres auquel appartient le sommet, force qui dérive de l'énergie de déformation.

Restant pour le modèle masse-tenseur, le problème des grandes déformations. [Pic02] a proposé un nouveau modèle déformable mettant en œuvre l'élasticité non-linéaire de St Venant-Kirchhoff.

2.2.4 Le modèle à barres

[Lyv03] a mené une étude sur l'approximation par des treillis de poutres des modèles mécaniques des milieux continus élastiques et non élastiques. Il a appliqué ces résultats à la modélisation des tissus mous. Il s'agit de représenter un organe par un réseau périodiques de barres non conformes au sens des éléments finis. Cette approche est appliquée à la simulation d'un modèle visco-hyperélastique de tissus vivants obtenu par couplage en parallèle d'amortisseurs linéaires aux barres élastiques. Ce modèle s'inscrit dans la même famille des modèles masse-liaison dont nous venons de voir le modèle masse-ressort et le modèle masse-tenseur.
Malgré la simplicité de ce modèle et la rapidité de calcul, ce modèle présente un inconvénient majeur dû à son maillage non conforme (au sens éléments finis). Un organe reconstruit en 3D d'après des images IRM doit subir une adaptation pour pouvoir être utilisé par ce modèle.

2.2.5 Comparaison des différents modèles

A l'heure actuelle, les deux principales méthodes numériques utilisées sont celles des éléments finis et des masse-ressort. Rappelons que la méthode des masse-tenseur est une restriction locale de la méthode des éléments finis.
La méthode masse-ressort a l'avantage d'être facile à implémenter. De plus, elle permet la résolution « temps réel », rendant possible le couplage avec un module haptique. Elle permet également de prendre facilement en compte l'anisotropie et l'hétérogénéité du matériau, en adaptant les cœfficients des modèles à la géométrie à simuler. Par contre, ne se basant pas sur des principes continus, les résultats sont moins fiables que pour les modèles éléments finis.
La méthode éléments finis (FEM) permet de résoudre le problème mécanique continu, elle offre ainsi l'avantage d'obtenir des résultats quantitativement corrects. Cependant, dans le cas de simulation d'opérations chirurgicales, le « temps réel »n'est pas accessible pour l'instant.

Comme nous l'avons présenté, l'organe subit pendant une opération des changements de topologie, c'est le cas de la découpe ; ce problème majeur pour la méthode éléments finis a été résolu par la méthode de masse-tenseur.

Les avantages et les inconvénients des différentes méthodes utilisées pour la simulation d'opérations chirurgicales sont reportés dans le Tableau 2.1.

2.3 Le modèle masse-ressort-amortisseur détaillé

L'idée du modèle « masse-ressort-amortisseur »est de représenter la matière par une discrétisation en un ensemble de petites masses, reliées entre elles par des liaisons type 'ressort' et/ou 'amortisseur' mises en série ou en parallèle (Figure 2.2). La liaison que nous avons utilisée dans l'implémentation du modèle masse-ressort est la liaison Kelvin-Voigt : un ressort et un amortisseur sont mis en parallèle.

TAB. 2.1 – Tableau récapitulatif des avantages et inconvénients des différentes méthodes

	Masse-Ressort	**Eléménts finis**	**Masse-Tenseur**
Facilité de mise en œuvre	⋆⋆⋆	⋆	⋆⋆
« Temps réel »	⋆⋆⋆	⋆	⋆⋆
Réalisme physique	⋆	⋆⋆⋆	⋆⋆
Changement de topologie	⋆⋆⋆	⋆	⋆⋆⋆
Grandes déformations	⋆⋆	⋆⋆	⋆
Grands déplacements	⋆⋆⋆	⋆	⋆

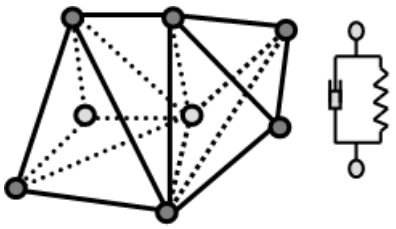

FIG. 2.2 – Modèle masse-ressort-amortisseur

Le principe d'un objet déformable de type masse-ressort repose sur l'équation fondamentale de la dynamique : $m_i\ddot{x}_i = \sum Forces_{appliquees}$.
En appliquant ce principe sur chaque masse, le bilan s'écrit alors :

$$m_i\ddot{x}_i = -\gamma_i\dot{x}_i + \sum_i g_{ij} + f_i \tag{2.1}$$

m_i est la masse d'un point, $x_i \in \mathbf{R}^3$ est sa position, le terme de droite est la somme des forces agissantes sur ce point. $-\gamma_i\dot{x}_i$ la force d'amortissement, ce terme a pour effet de simuler un comportement viscoélastique. g_{ij} est la force exercée sur la masse i par le ressort liant les masses i et j, et f_i est la somme des forces externes (gravité ou autres forces) appliquées sur la masse i.

Ce bilan s'écrit sous forme matricielle :

$$M\ddot{X} + C\dot{X} + K(X) = F \tag{2.2}$$

Où M, C et K sont respectivement les matrices 3N x 3N de Masses, amortissements et des raideurs. N étant le nombre des sommets du maillage.
Le problème peut être ramené à la résolution d'un système différentiel de premier ordre :

$$\dot{V} = M^{-1}(-CV - K(X) + F) \qquad \dot{X} = V \tag{2.3}$$

Où V est le vecteur de vitesse de l'ensemble des masses.

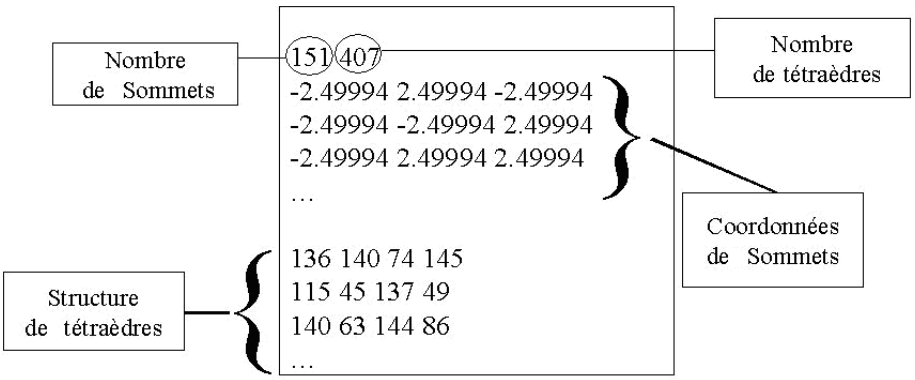

FIG. 2.3 – Structure du fichier de maillage utilisé

2.3.1 Maillage

La topologie d'un modèle masse-ressort est un problème peu étudié. La première contrainte justifiant le choix d'une topologie correcte est liée à la modélisation discrète du solide. En théorie il est nécessaire que la répartition des sommets du maillage et les masses associées ne modifient pas le centre de masse ni la matrice d'inertie du solide que l'on cherche à simuler.
Le maillage que nous avons utilisé est fourni par le logiciel Forge3 de résolution numérique des éléments finis[1].
Il s'agit d'un fichier contenant le nombre des sommets et le nombre des tétraèdres ainsi que les positions initiales des ces sommets et les indices des sommets formant les tétraèdres comme le montre la figure 2.3.
Pour pouvoir exploiter ces données, il est nécessaire d'extraire de ce fichier une structure de données permettant une meilleure interprétation du maillage, en particulier avec les points de la surface, puisqu'une interaction ou une collision se fait avec les sommets de surface. Il est également nécessaire d'extraire les liaisons associées à chaque masse pour pouvoir calculer la force.
Chaque tétraèdre a au maximum quatre tétraèdres voisins (un tétraèdre sur chaque facette). Nous avons donc extrait pour chaque tétraèdre les indices des quatre tétraèdres voisins, un -1 étant utilisé à la place de l'indice s'il s'agit d'une facette extérieure.
L'extraction de la surface extérieure du maillage se fait en récupérant tout tétraèdre qui a moins de quatre voisins. Les points de surfaces sont utiles pour l'affichage de l'objet et pour le traitement de la détection de collision.
Après l'extraction des points de surfaces nous extrayons les liaisons associées à chaque sommet, la figure 2.4 nous montre la structure de données que nous obtenons après extraction de la surface et des liaisons associées.

[1]http ://www-cemef.cma.fr

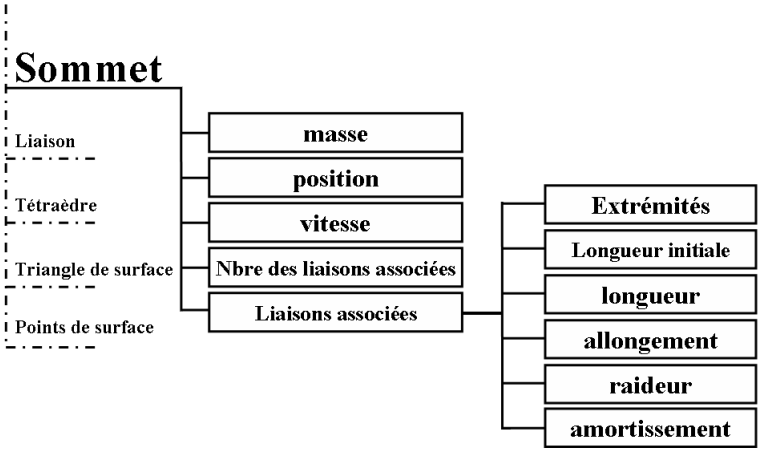

FIG. 2.4 – Structure de données

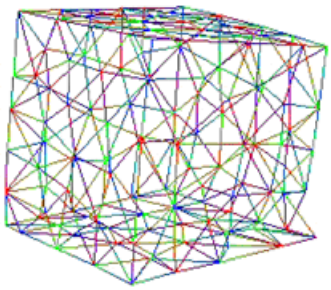

FIG. 2.5 – Gravité appliquée au cube sans homogénéité de masse volumique

2.3.2 Homogénéité des masses

Nous avons testé notre implémentation sur un cube du maillage tétraédrique, ce maillage n'est pas homogène. Nous avons placé ce cube à une hauteur du sol et nous lui avons appliqué un champ de gravité. La figure 2.5 nous montre la réaction de la chute du cube mis dans un champ de gravité. Nous remarquons sur cette figure la non symétrie du comportement du cube, cette non symétrie est due à la non homogénéité du maillage et l'utilisation des masses égales.
Nous avons pensé alors à ne pas utiliser les mêmes valeurs des masses pour tous les sommets. Il doit y avoir une relation entre la répartition des sommets du maillage et les masses associées. L'idée est de prendre en compte la masse volumique de l'organe et de calculer la masse d'un tétraèdre k qui dépend alors de son volume Vol_k et ensuite de répartir cette masse sur les quatre sommets du tétraèdre. Alors la masse de chaque sommet dépend de tous les tétraèdres contenant ce même point2.5.

$$m = \rho Vol \tag{2.4}$$

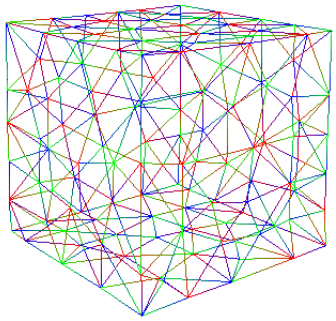

FIG. 2.6 – Gravité appliquée au cube avec homogénéité de masse volumique

$$m_i = \rho \sum_k \frac{1}{4} Vol_k \tag{2.5}$$

La figure 2.6 représente le même cube soumis au champ de gravité avec homogénéité de masse. La masse globale du cube est la même que dans le premier essai et les caractéristiques des liaisons sont les mêmes. Nous trouvons que cette procédure d'homogénéité de masse a amélioré clairement le rendu du comportement.

2.3.3 Identification des raideurs

Le comportement des réseaux masses-ressorts est directement lié au maillage utilisé. Ce maillage est souvent irrégulier, nous ne pouvons pas donc affecter les mêmes raideurs pour tous les ressorts. Van Gelder présente une méthode pour identifier les raideurs à donner aux ressorts [Gel98] dans un maillage triangulé en deux dimensions. En simplifiant considérablement le modèle continu de la matière et en se basant sur les éléments finis, Van Gelder calcule la raideur k_c d'un ressort en fonction de la géométrie des deux triangles incidents à ce dernier selon cette formule :

$$k_c = \frac{E \sum S_e}{|c^2|} \tag{2.6}$$

Se est la surface du triangle incident. c est la longueur de l'arête. E est un coefficient identique pour tous les ressorts, c'est le module d'Young. Nous étendrons alors cette équation pour un maillage tétraédrique. Nous ne développons pas ici les détails théoriques. Comme résultat, nous obtenons une expression qui fixe la constante de rigidité k_c de chaque ressort :

$$k_j = \frac{E \sum_e V_e}{|c^2|} \tag{2.7}$$

Où les V_e sont les volumes des tétraèdres qui ont comme arête le ressort j dont nous calculons la raideur.

	Number of Points	Number of Springs	Number of Tetrahedra
Irregular Cube (a)	151	688	407
Cube_1 (b)	8	18	5
Cube 5x5x5 (c)	125	540	320
Cube 10x10x10 (d)	1000	5130	5645

2.3.4 Test rhéologique et essai sur le modèle numérique

Pour tester le modèle masse-ressort, nous avons procédé à une analyse du comportement mécanique du modèle virtuel de corps mous que nous avons développé. Il s'agit des expérimentations virtuelles comme celles présentées pour les tests rhéologiques des organes réels.
Nous présentons des résultats quantitatifs de l'étude statique de notre modèle virtuel après déformation (compression, indentation), et des résultats qualitatifs de l'étude dynamique.
Nous nous sommes intéressés à la modélisation d'un corps dynamique, c'est-à-dire que les caractéristiques de notre matériau virtuel sont à la fois statiques et dynamiques.
Toutefois dans l'optique d'une analyse comparative (quantitative) de notre modèle virtuel avec des modèles de la mécanique des milieux continus, nous nous limitons dans un premier temps aux caractéristiques statiques, c'est-à-dire à l'étude des solutions stationnaires (asymptotiques) des déformations. Nous ne présentons des résultats dynamiques que pour des considérations qualitatives.
Pour une même enveloppe, nous pouvons avoir différents maillages. Tous les maillages que nous utilisons sont des maillages tetraédriques. Pour cette analyse nous avons utilisé quatre maillages différents pour un cube de 50 mm de coté : un maillage irrégulier, provenant d'un logiciel commercial, trois autres maillages que nous avons réalisés permettant de représenter ce même cube selon différentes résolutions (figure 2.7).
Valeurs numériques
Densité ρ = 1000kg/m^3 (0.001g/mm^3)
Module d'Young : E=0.02 MPa
Masses m_i (eq 2.5)
Masse totale pour chaque cube = 125g(0.125Kg)
Cœfficient d'amortissement : 0.001 N.s/m
Gravité g=9.8 m/s^2
Δt (pas de temps pour l'intégration numérique)= 0.1ms.

Nous avons réalisé une compression des différents cubes par un plan horizontal. La force exercée résulte de la somme des forces de réaction sur chaque point de surface. nous appliquons une compression de 12 mm.

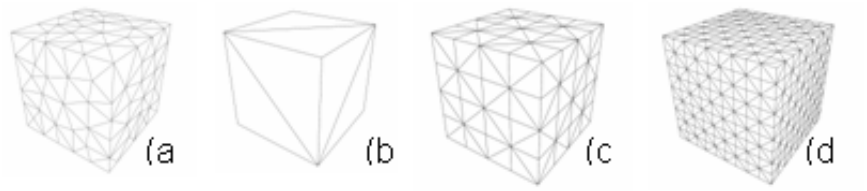

FIG. 2.7 – Différents types de maillage

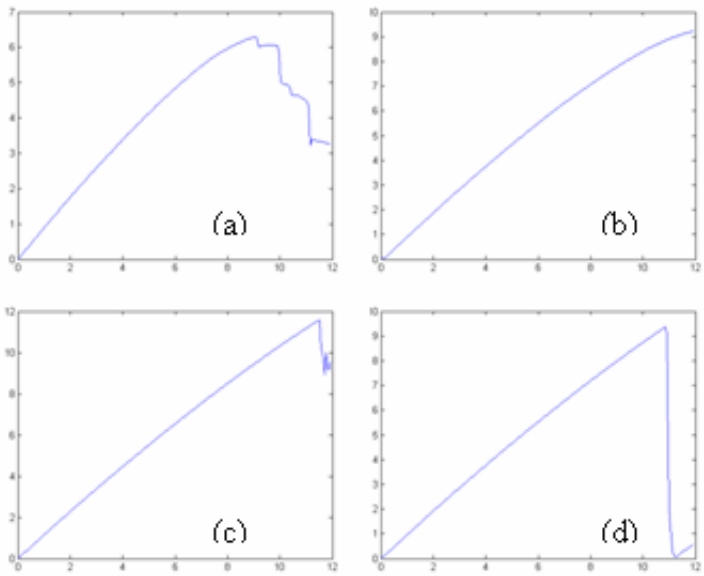

FIG. 2.8 – Force-displacement curves (a) :irregular cube ; (b) :cube 1 ; (c) :cube 5x5x5x ; (d) :cube 10x10x10

Mesure du module d'Young

$$E = \frac{contrainte}{deformation} = \frac{force/surface}{deplacement/longueurinitiale} \tag{2.8}$$

$E_{cubeirrgulier} = 0.0178$ MPa
$E_{Cube1} = 0.0221$ MPa
$E_{Cube5x5x5x} = 0.0224$ MPa
$E_{Cube10x10x10} = 0.0189$ MPa

Nous observons l'obtention à 10% pré du module d'Young déjà imposé. Nous remarquons que le module d'Young mesuré expérimentalement subit une petite variation d'un maillage à un autre.

2.3.5 Limitations du modeles

Le modèle masse-ressort atteint vite ses limites, nous remarquons l'influence du maillage sur les forces de réactions suite à une compression. Nous remarquons aussi lors de ces essais virtuels que ce modèle est complètement compressible car les cubes n'a pas subi aucune déformation dans le sens perpendiculaire à celui de la compression ce qui n'est pas le cas des organes qui sont constitués d'une grande proportion d'eau.

Cette conclusion ne nous semble pas surprenante, car le modèle masse ressort ne reflète pas le comportement physique réel des organes déformables. Nous nous sommes toutefois servi de la simplicité de ce modèle pour réaliser une plateforme de simulation, regroupant en plus de l'intégration numérique que nous abordons ci-dessous, la détection de collision, la gestion de collision et la découpe que nous traitons dans le chapitre 4.

2.4 Intégration numérique

L'intégration numérique est un problème important des modèles déformables. Le choix du pas de temps et de la méthode d'intégration revêt une grande importance dans le comportement des modèles de type masse-ressort. Il y a deux problèmes liés à l'intégration numérique. Le premier concerne le choix du pas de temps Δt. Le second problème est lié aux erreurs de calcul en fonction de la méthode d'intégration utilisée.
On peut employer différentes méthodes d'intégration, toutes demandent d'adapter le pas de temps d'intégration à la valeur des accélérations. Si le pas de temps est trop petit, on perdra du temps de calcul alors que s'il est trop grand le système divergera. Nous distinguons deux types d'intégration : l'intégration explicite et l'intégration implicite.

2.4.1 Intégration explicite

Les méthodes d'intégration explicites sont les plus simples et s'appuient directement sur le développement de Taylor.
La plus simple d'entre elles est la méthode d'Euler, qui n'utilise que des développements

Méthode d'intégration	Equation
Euler	$X = X_0 + h * f(X_0)$
Runge Kutta d'ordre 2	$k_1 = f(X_0) * h$ $k_2 = f(X_0 + \frac{1}{2}k_1) * h$ $X = X_0 + k_2$
Runge Kutta d'ordre 4	$k_1 = f(X_0) * h$ $k_2 = f(X_0 + \frac{1}{2}k_1) * h$ $k_3 = f(X_0 + \frac{1}{2}k_2) * h$ $k_4 = f(X_0 + k_3) * h$ $X = X_0 + k_4$

FIG. 2.9 – Méthodes d'intégration explicites

limités d'ordre 1 pour écrire :

$$v_{t+dt} = v_t + \frac{F_t}{m}dt$$
$$p_{t+dt} = p_t + v_t dt$$

Simple, elle offre néanmoins parfois de bons résultats.

En faisant dépendre la nouvelle position de la nouvelle vitesse et non plus de celle du pas précédent, on obtient le schéma d'**Euler modifié** que nous qualifions comme une méthode **semi implicite**. En pratique la méthode d'Euler modifiée est plus stable que la methode précédente. Il a été démontré par Provost dans sa thèse [Pro97] que cette méthode donne souvent des résultats supérieurs à ceux de Runge-Kutta 2 (Figure2.9).

$$v_{t+dt} = v_t + \frac{F_t}{m}dt$$

$$p_{t+dt} = p_t + v_{t+dt}dt$$

Nous avons implémenté cette méthode d'intégration pour le modèle masse-ressort ainsi que pour le modèle masse-liaison étendu présenté dans le chapitre 3.
On remarque que dans cette méthode les forces (ou accélérations) au temps t contribuent

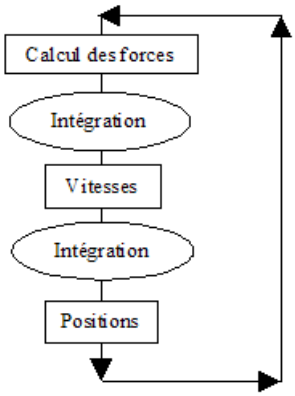

FIG. 2.10 – Boucle d'intégration

aux vitesses au temps $t+dt$. Un tel système se comporte bien pour de petits intervalles de temps mais il diverge rapidement lorsqu'on augmente le pas de temps. Un autre schéma qui est mieux adapté à ce problème est celui de l'intégration implicite.

2.4.2 Intégration implicite

L'intégration implicite fut remise au goût du jour pour la simulation de déformation par Baraff et Witkin dans [BW98]. Elle donne d'excellents résultats pour des raideurs élevées que l'intégration explicite ne sait pas gérer, ou au prix d'un pas de temps rédhibitoire.
L'idée consiste à ne plus trouver la nouvelle position en fonction des forces connues à un instant, mais à chercher la position où les forces à $t+dt$ auraient conduits, si on les avait intégrées en partant de la position courante. Plutôt qu'un bond en aveugle dans le futur, on cherche un état où les forces sont compatibles avec la position. Mathématiquement, Euler deviendrait :

$$v_{t+dt} = v_t + \frac{F_{t+1}}{m}dt$$
$$p_{t+dt} = p_t + v_{t+1}dt$$

Cela demande donc de connaître F_{t+dt} pour trouver p_{t+dt} , ce qui parait impossible puisque la force dépend elle même de la position. Donc pour implémenter ce système d'équations, on doit calculer F_{t+dt} sans connaitre les positions des masse au temps $t+dt$. En écrivant l'approximation de 1er ordre :

$$F_{t+dt} = F_t + \frac{\partial F}{\partial x}\Delta x_{t+dt}$$

Où

$$\Delta x_{t+dt} = x_{t+dt} - x_t$$

On écrit $H = \frac{\partial F}{\partial x}$ où H est la matrice hessienne du système. On traite cette matrice en bloc : $H \in \Re^{3n \times 3n}$, où n est le nombre de masses du système. Il s'ensuit que chaque élément H_{ij} est une matrice 3x3.
Nous pouvons alors écrire :

$$\Delta_{t+dt} v = (I - \frac{dt^2}{m} H)^{-1} (F_t + dt H v_t) \frac{dt}{m}$$

Où I est la matrice identité.
Pour pouvoir résoudre $\Delta_{t+dt} v$ il faut calculer la matrice H, ensuite inverser de la matrice $(I - \frac{dt^2}{m} H)$ à chaque pas de temps. Nous pouvons donc calculer $\Delta_{t+dt} x = (v_t + \Delta_{t+dt} v) dt$ et ainsi obtenir la nouvelle position x_{t+dt} du système.

2.4.3 La méthode d'Euler retrograde

Nous avons utilisé une méthode de différentiation de type Euler implicite. on trouve une explication détaillée dans les ouvrages de référence [Cro92] et [Dem96].
[Cro92] montre que cette méthode est stable. On pose, pour un pas dt donné, l'interpolation de la dérivée à l'ordre r :

$$\dot{x}_i(t) = \frac{1}{dt} \sum_{s=0}^{r} \alpha_s x_i^s$$

$$\ddot{x}_i(t) = \frac{1}{dt^2} \sum_{s=0}^{r} \alpha_s \dot{x}_i^s(t) = \frac{1}{dt^2} \sum_{s=0}^{r} \beta_s x_i^s(t)$$

avec x_i^s le vecteur position au temps $t - sdt$:

$$x_i^s = x_i(t - sdt)$$

Prenons le cas de notre étude :

- *Premier ordre :*

$$\dot{x}_{ik} = \frac{x_{ik}(t) - x_{ik}(t - dt)}{dt}$$

- *Deuxième ordre :*

$$\dot{x}_{ik} = \frac{-3x_{ik}(t) + 4x_{ik}(t - dt) - x_{ik}(t - 2dt)}{dt}$$

d'où

$$\ddot{x}_{ik} = \frac{9x_{ik}(t) - 24x_{ik}(t - dt) + 22x_{ik}(t - 2dt) - 8x_{ik}(t - 3dt) + x_{ik}(t - 4dt)}{4dt^2}$$

Ce type de discrétisation en temps introduit dans l'équation 2.1 :

$$x_{ik} = -\frac{1}{\beta_0} \sum_{s=i}^{r} \beta_s x_{ik}^s + \frac{dt^2}{\beta_0 m_i} (-\gamma_i \dot{x}_i + \sum_i g_{ij} + f_i) \tag{2.9}$$

Ce système est de la forme :

$$X(t) = G(X(t)),$$

avec G non linéaire.

2.4.4 Résolution par la méthode du point fixe (Point fixe de Banach)

Nous utilisons la méthode du point fixe de Banach pour résoudre 2.9, préférée à la méthode de Newton moins performante en terme de calcul.

La méthode de Newton demande le calcul et l'inversion de la matrice Jacobienne de la fonction G définie plus haut. Or son calcul et son inversion demande la mise en oeuvre d'un algorithme de stockage et de résolution de système linéaire de grande taille très coûteux en temps (même si dans notre cas la matrice est creuse). Ce type de méthode n'est pas compatible avec la contrainte « temps réel »imposée en simulation de chirurgie.

2.5 Conclusion

Nous venons de présenter les différents modèles déformables ainsi que les problématiques d'une bonne modélisation. Le modèle **masse-ressort** que nous avons développé, est encore très présent car il permet la prise en compte du retour de force (via un calcul « temps réel »). Il permet ainsi au chirurgien de se familiariser avec l'environnement particulier des opérations mini-invasives, malgré les insuffisances et les limitations que nous venons de détailler. Ce modèle nous a permis d'approfondir notre connaissance du besoin de la simulation chirurgicale. Sa simplicité de mise en œuvre et sa rapidité nous ont permis de valider la méthode d'intégration et de travailler en parallèle sur l'interaction avec les corps déformables (Chapitre 4).

Dans le chapitre suivant nous détaillons notre approche, issue de la Mécanique des Milieux Continus, pour la modélisation temps réel des corps déformables. Elle repose sur le modèle des matériaux de Saint Venant Kirchhoff qui correspondent à une classe de matériaux hyperélastiques.

Chapitre 3

Modèle viscoélastique non-linéaire de déformation

Dans ce chapitre nous détaillons une nouvelle approche pour la modélisation temps réel des corps déformables. Il s'agit d'une méthode physique, qui offre la possibilité de paramétrer le modèle à partir de mesures expérimentales, et de construire une solution générique applicable à une grande variété de matériaux. Il s'agit d'une approche issue de la Mécanique des Milieux Continus ; c'est une synthèse entre deux presentations répandues : les masses-ressorts et les masses tenseurs. Elle repose sur le modèle des matériaux de Saint Venant Kirchhoff qui correspondent à une classe de matériaux hyperélastiques.

3.1 Introduction

Comme nous l'avons déjà mentionné, à l'heure actuelle, les deux principales méthodes numériques utilisées sont celles des éléments finis et des masses-ressort.
Le modèle **masse-ressort** est encore très présent car il permet la prise en compte du retour de force (via un calcul « temps réel »). Il permet ainsi au chirurgien de se familiariser avec l'environnement particulier des opérations mini-invasives, malgré les insuffisances et les limitations que nous venons de détailler dans 2.3.5.
Le modèle **éléments finis** est de plus en plus employé de par la précision des résultats auxquels il conduit. Mais la lenteur relative du modèle ne lui permet pas aujourd'hui de simuler une opération en « temps réel ». La notion du « temps réel » indispensable pour la simulation médicale, n'empêchera pas l'utilisation du modèle éléménts finis surtout par des « biomécaniciens »qui s'intéressent à la caractérisation précise des lois de comportement des tissus biologiques [Mar05]. La méthode des éléments finis démontre ses possibilités aussi dans le cadre de modélisations « off-line », comme par exemple la planification d'opérations chirurgicales ne nécessitant pas une réponse immédiate du logiciel. On peut alors utiliser à la fois un modèle précis (car complet) associé à une loi de comportement réaliste et dont les paramètres ont été identifiés précisément. L'approche « masse-tenseur »est une simplification de la méthode éléments-finis, il s'agit d'une résolution locale et itérative.
L'approche que nous proposons pour la simulation temps-réel des corps déformables est une méthode physique, qui offre la possibilité de paramétrer le modèle à partir de mesures expérimentales, et de construire une solution générique applicable à une grande variété de matériaux. Elle repose sur un modèle continu de la matière, ce qui la rend indépendante de la topologie du maillage et de la forme de l'objet. Elle permet aussi de simuler des changements de topologie, indispensables pour la chirurgie virtuelle. Elle offre en outre une grande flexibilité de mise en oeuvre. Il s'agit d'une approche issue de la Mécanique des Milieux Continus ; c'est une synthèse entre deux présentations répandues : les masses-ressorts et les masses tenseurs.
Ce modèle combine la simplicité du premier avec le réalisme du second. L'aspect « temps-réel »restera le point fort de cette approche, il s'agit d'une complexité linéaire en fonction de nombre de tétraèdre. Nous avons utilisé cette approche pour modéliser un comportement viscoélastique non linéaire. Il s'agit des matériaux de Saint Venant Kirchhoff qui correspondent à une classe de matériaux hyperélastiques. Avant de développer cette approche, nous rappelons les bases théoriques sur lesquelles nous nous appuyons pour la formulation de l'approche.

3.2 Rappels de la Mécanique des Milieux Continus

Dans cette section nous représentons quelques notions de base de la mécanique des milieux continus. Cette partie n'a pas pour vocation de faire un cours de mécanique, elle est basée sur les sources suivantes [Chev96] [Sal02].

3.2.1 Rappel des tenseurs

Soit Ω_0 la configuration initiale avant déformation. On note x_0 un point du matériau en position initiale, x le même point en position déformée.

$$x = \Phi(x_0) \tag{3.1}$$

Lors de la déformation du matériau, pour décrire les déformations locales autour d'un point solide, on introduit un vecteur élémentaire dx qui s'exprime par :

$$dx = dx_0 + \frac{\partial u}{\partial x_0} dx_0 = (I + grad(u))dx_0 = F dx_0 \tag{3.2}$$

avec $x = x_0 + u(x_0, t)$, u étant le déplacement et F le tenseur gradient de la transformation, d'où

$$F = \nabla\Phi \tag{3.3}$$

On en déduit le tenseur de Cauchy-Green droit (ou tenseur des dilatations de la transformation) Φ la matrice :

$$C = \nabla\Phi^T \nabla\Phi = F^T F \tag{3.4}$$

Le tenseur de déformation de Green-Lagrange (ou tenseur de déformation de Green-Saint Venant) est défini par :

$$\boldsymbol{E} = \frac{1}{2}(C - I) = \frac{1}{2}\big(grad(u) + (grad(u))^T + (grad(u))^T (grad(u))\big) \tag{3.5}$$

Notons que $\boldsymbol{E}$ a les mêmes directions principales que C mais ne prend pas en compte le mouvement du corps rigide, ce qui nous permet de couvrir le cas des grandes déformations. On remarque que le tenseur de déformation est toujours symétrique.

Nous introduisons aussi le **tenseur des contraintes de Cauchy**σ produite par les déformations. La relation entre le tenseur des contraintes et le tenseur des déformations caractérise le comportement mécanique du matériau. Cette relation est appelée loi constitutive du matériau.

3.2.2 Rappel de la loi de Hooke

Dans le cas des petites déformations, on linéarise le tenseur de déformation en négligeant les termes du deuxième ordre du tenseur $\boldsymbol{E}$:

$$\varepsilon = \frac{1}{2}(grad(u) + (grad(u))^T) \tag{3.6}$$

En pratique on estime que si la déformation est inférieur ou égale à 10%, la condition de linéarité est vérifiée [Cou83][Wel99]. Le terme *linéaire* signifie que la relation entre contraintes et déformations et du premier ordre. La relation qui relie les contraintes aux déformations est dite loi de comportement. Prenons dans un premier temps les cas des petites déformations, nous utilisons alors la loi de Hooke.
Un milieu élastique linéaire isotrope, ce qui signifie que les propriétés sont les mêmes

dans toutes les directions, est caractérisé par une énergie de déformation volumique W_E quadratique définie positive [Lai93]. Dans le cas d'un milieu isotrope, W_E ne dépend que des invariants de ε. Elle se met sous la forme :

$$W_E = \frac{1}{2}(\lambda(Tr(\varepsilon))^2 + 2\mu Tr(\varepsilon^2) \tag{3.7}$$

λ et μ sont les coeffcients de Lamé, I la matrice identité et $Tr(\varepsilon)$ la trace de ε.
La loi de comportement dans le cadre de l'élasticité linéaire isotrope s'ecrit donc :

$$\sigma = \frac{\partial W_E}{\partial \varepsilon} \tag{3.8}$$

Elle est définie par une fonction linéaire reliant la contrainte à la déformation, c'est la loi de Hooke :

$$\sigma = \lambda Tr(\varepsilon)I + 2\mu\varepsilon \tag{3.9}$$

La forme duale s'écrit en inversant la relation de sorte à exprimer ε en fonction de σ :

$$\varepsilon = \frac{\nu}{E}Tr(\sigma)I + \frac{1+\nu}{E}\sigma \tag{3.10}$$

E est appelé module d'Young, il caractérise la rigidité du matériau, ν est le coefficient de Poisson, il rend compte de la compressibilité du matériau (pour $\nu = 0.5$, le matériau est parfaitement incompressible). Il est tout à fait équivalent de caractériser un matériau élastique linéaire isotrope par ses deux coefficients de Lamé ou par son module de Young et son coefficient de Poisson. Les deux expressions reliant les coefficients de Lamé au module de Young et au coefficient de Poisson sont :

$$\lambda = \frac{\nu E}{(1+\nu)(1-2\nu)} \qquad \mu = \frac{E}{2(1+\nu)} \tag{3.11}$$

$$E = \frac{\mu(3\lambda + 2\mu)}{\lambda + \mu} \qquad \nu = \frac{\lambda}{2(\lambda+\mu)} \tag{3.12}$$

3.2.3 Elasticité non linéaire : L'hyperélasticité

Lors des grandes déformations, le comportement d'un matériau élastique n'est pas linéaire, il est hyperélastique. Dans le cas de l'élasticité non linéaire, deux autre grandeurs importantes viennent s'ajouter, il s'agit des deux tenseurs de Piola-Kirchhoff S. Ils sont reliés au tenseur de Cauchy par :
Le premier tenseur de Piola-Kirchhoff :

$$P = FS \tag{3.13}$$

et S est le second tenseur de Piola-Kirchhoff :

$$S = JF^{-1}\sigma^T(F^{-1}) \quad ou \quad \sigma = J^{-1}FSF^T \tag{3.14}$$

avec F le tenseur des transformations, $J = detF$ le déterminant jacobien et σ le tenseur des contraintes de Cauchy.
Les matériaux hyperélastiques sont caractérisés par l'existence d'une énergie de déformation W ne dépendant que de la déformation actuelle F. Nous pouvons écrire cette énergie comme une fonction du tenseur des dilatations de Cauchy Green droit C. Le second tenseur de Piola-Kirchhoff S s'obtient en dérivant l'énergie de déformation W par rapport à C :

$$S = 2\frac{\partial W}{\partial C} \tag{3.15}$$

Dans le cadre des matériaux isotropes, W est indépendante du repérage choisi dans la configuration de référence. [Ger73] montre alors que W est une fonction isotrope de $\boldsymbol{E}$.
Il existe plusieurs modèles d'hyperélasticité, de complexités différentes et de domaines de validité différents. Nous renvoyons le lecteur à [Ver97] pour une liste détaillée des différents modèles existants. Nous nous intéresserons au modèle de Saint Venant Kirchhoff souvent utilisé pour la modélisation des corps déformables, où :

$$W = \frac{1}{2}(\lambda(Tr(\boldsymbol{E}))^2 + 2\mu Tr(\boldsymbol{E}^2) \tag{3.16}$$

ainsi que :

$$S = \lambda Tr(\boldsymbol{E})I + 2\mu\boldsymbol{E} \tag{3.17}$$

rappelant que :

$$\boldsymbol{E} = \frac{1}{2}(C - I) \tag{3.18}$$

3.3 La méthode Masse liaison étendue

Notre modèle se base sur un maillage tétraédrique du matériau étudié. Le tétraèdre est la primitive la plus simple, permettant de mailler des formes quelconques sans introduire d'anisotropie topologique. Nous adoptons une interpolation linéaire sur les éléments finis (approximations P1 des éléments finis).

En chaque nœud i du maillage sont affectées des masses m_i ; et entre les nœuds, des forces élastiques de rappel F_i de la forme :

$$F_i = \sum_{T_j} \left(f_{i,j,1}(L_j) * v_{j,1} + f_{i,j,2}(L_j) * v_{j,2} + f_{i,j,3}(L_j) * v_{j,3}\right) \tag{3.19}$$

En notant T_j les tétraèdres incidents au nœud i, $v_{j,1-3}$ les trois arêtes du tétraèdre T_j incidentes au nœud i, et L_j le vecteur des carrés des normes des 6 arêtes du tétraèdre T_j :

$$L_j = \left(l_{j,1}^2, l_{j,2}^2, l_{j,3}^2, l_{j,4}^2, l_{j,5}^2, l_{j,6}^2\right) \tag{3.20}$$

Aux forces élastiques de rappel peuvent être ajoutées des forces de frottement modélisant le comportement visqueux du matériau.

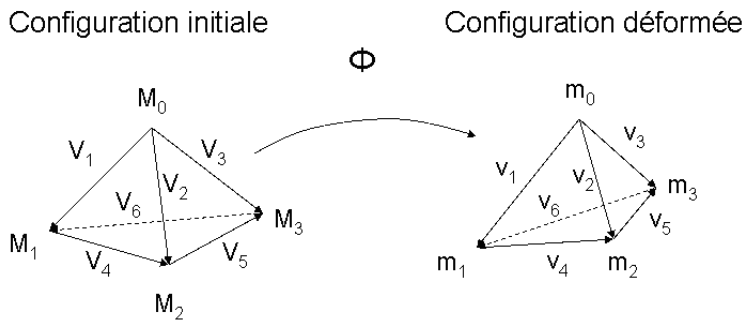

FIG. 3.1 – configurations initiale et déformée

Matériaux de Saint Venant-Kirchhoff
Dans le cas des matériaux de Saint Venant-Kirchhoff, les fonctions $f_{i,j,k}$ sont linéaires en L_j. Pour un tétraèdre donné, les coefficients de ces fonctions ne dépendent que de la configuration initiale du tétraèdre et des coefficients de Lamé λ et μ du matériau.

Avantages du modèle
Ce modèle est une extension du modèle Masses-Ressorts, pour lequel les forces élastiques incidentes au nœud i s'écrivent :

$$F_i = \sum_{T_j} \left(\left(\frac{l_{j,1} - l^0_{j,1}}{l_{j,1}} \right) * v_{j,1} + \left(\frac{l_{j,2} - l^0_{j,2}}{l_{j,2}} \right) * v_{j,2} + \left(\frac{l_{j,3} - l^0_{j,3}}{l_{j,3}} \right) * v_{j,3} \right) \tag{3.21}$$

Il est mathématiquement identique à la représentation Masses-Tenseurs, avec une formulation et une complexité algorithmique plus simples. Pour les calculs lors des itérations, ce modèle ne fait pas apparaître les positions initiales des nœuds ou les positions relatives à la position initiale, mais seulement les positions relatives aux autres nœuds dans une configuration donnée.

3.3.1 Configurations initiale et déformée

On définit les vecteurs des arêtes du tétraèdre de la façon suivante :

$$\text{Initiale : } \begin{cases} V_1 = M_1 - M_0 \\ V_2 = M_2 - M_0 \\ V_3 = M_3 - M_0 \\ V_4 = M_2 - M_1 \\ V_5 = M_3 - M_2 \\ V_6 = M_1 - M_3 \end{cases} \qquad \text{Déformée : } \begin{cases} v_1 = m_1 - m_0 \\ v_2 = m_2 - m_0 \\ v_3 = m_3 - m_0 \\ v_4 = m_2 - m_1 \\ v_5 = m_3 - m_2 \\ v_6 = m_1 - m_3 \end{cases} \tag{3.22}$$

On définit la matrice ayant pour colonnes les trois premiers vecteurs des arêtes du tétraèdre :

$$\text{Initiale : } V = (V_1, V_2, V_3) \qquad \text{Déformée : } v = (v_1, v_2, v_3) \tag{3.23}$$

Les autres vecteurs des arêtes peuvent s'écrire en fonction des trois premiers vecteurs :

$$\text{Initiale : } \begin{cases} V_4 = V_2 - V_1 \\ V_5 = V_3 - V_2 \\ V_6 = V_1 - V_3 \end{cases} \qquad \text{Déformée : } \begin{cases} v_4 = v_2 - v_1 \\ v_5 = v_3 - v_2 \\ v_6 = v_1 - v_3 \end{cases} \tag{3.24}$$

Par définition on utilise des tétraèdres initiaux non dégénérés, la matrice V est donc inversible et on note B son inverse : $B = V^{-1}$

Longueurs des arêtes L

On note $l_{1-0,..6-0}$ les longueurs des arêtes en position initiale et l_{1-6} les longueurs des arêtes en position déformée. On note L le vecteur des longueurs des arêtes au carré en position déformée, L_0 en position initiale, ΔL l'écart entre position initiale et déformée :

$$\text{Initiale : } L_0 = \begin{pmatrix} l_{1_0}^2 \\ l_{2_0}^2 \\ l_{3_0}^2 \\ l_{4_0}^2 \\ l_{5_0}^2 \\ l_{6_0}^2 \end{pmatrix} \text{Déformée : } L = \begin{pmatrix} l_1^2 \\ l_2^2 \\ l_3^2 \\ l_4^2 \\ l_5^2 \\ l_6^2 \end{pmatrix} \text{Ecart : } \Delta L = L - L_0 = \begin{pmatrix} l_1^2 - l_{1_0}^2 \\ l_2^2 - l_{2_0}^2 \\ l_3^2 - l_{3_0}^2 \\ l_4^2 - l_{4_0}^2 \\ l_5^2 - l_{5_0}^2 \\ l_6^2 - l_{6_0}^2 \end{pmatrix} \tag{3.25}$$

L peut s'écrire en fonction des vecteurs v_{1-3} soit en fonction des sommets m_{0-4} :

$$L_0 = \begin{pmatrix} \|v_1\|^2 \\ \|v_2\|^2 \\ \|v_3\|^2 \\ \|v_2 - v_1\|^2 \\ \|v_3 - v_2\|^2 \\ \|v_1 - v_3\|^2 \end{pmatrix} \qquad L = \begin{pmatrix} \|m_1 - m_0\|^2 \\ \|m_2 - m_0\|^2 \\ \|m_3 - m_0\|^2 \\ \|m_2 - m_1\|^2 \\ \|m_3 - m_2\|^2 \\ \|m_1 - m_3\|^2 \end{pmatrix} \tag{3.26}$$

Dérivées de L par rapport aux sommets du tétraèdre

On a :

$$\begin{cases} l_1^2 = (M_1 - M_0)^T * (M_1 - M_0) \\ l_2^2 = (M_2 - M_0)^T * (M_2 - M_0) \\ l_3^2 = (M_3 - M_0)^T * (M_3 - M_0) \\ l_4^2 = (M_2 - M_1)^T * (M_2 - M_1) \\ l_5^2 = (M_3 - M_2)^T * (M_3 - M_2) \\ l_6^2 = (M_1 - M_3)^T * (M_1 - M_3) \end{cases} \tag{3.27}$$

On peut déterminer la dérivée de chacun de ces termes par rapport aux positions m_{0-4} et l'exprimer comme la dérivée du vecteur L :

$$\text{Dérivées par rapport à } m_0 : \begin{cases} \frac{\partial l_1^2}{\partial m_0} = -2(M_1 - M_0)^T = -2v_1^T \\ \frac{\partial l_2^2}{\partial m_0} = -2(M_2 - M_0)^T = -2v_2^T \\ \frac{\partial l_3^2}{\partial m_0} = -2(M_3 - M_0)^T = -2v_3^T \\ \frac{\partial l_4^2}{\partial m_0} = 0 \\ \frac{\partial l_5^2}{\partial m_0} = 0 \\ \frac{\partial l_6^2}{\partial m_0} = 0 \end{cases} \tag{3.28}$$

$$Soit : \frac{\partial L}{\partial m_0} = 2 * \begin{pmatrix} -v_1^T \\ -v_2^T \\ -v_3^T \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} -2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} * v_1^T + \begin{pmatrix} 0 \\ -2 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} * v_2^T + \begin{pmatrix} 0 \\ 0 \\ -2 \\ 0 \\ 0 \\ 0 \end{pmatrix} * v_3^T \tag{3.29}$$

$$\text{Dérivées par rapport à } m_1 : \begin{cases} \frac{\partial l_1^2}{\partial m_1} = 2(M_1 - M_0)^T = 2v_1^T \\ \frac{\partial l_2^2}{\partial m_1} = 0 \\ \frac{\partial l_3^2}{\partial m_1} = 0 \\ \frac{\partial l_4^2}{\partial m_1} = -2(M_2 - M_1)^T = 2(v_1 - v_2)^T \\ \frac{\partial l_5^2}{\partial m_1} = 0 \\ \frac{\partial l_6^2}{\partial m_1} = 2(M_1 - M_3)^T = 2(v_1 - v_3)^T \end{cases} \tag{3.30}$$

$$Soit : \frac{\partial L}{\partial m_1} = 2 * \begin{pmatrix} v_1^T \\ 0 \\ 0 \\ (v_1 - v_2)^T \\ 0 \\ (v_1 - v_3)^T \end{pmatrix} = \begin{pmatrix} 2 \\ 0 \\ 0 \\ 2 \\ 0 \\ 2 \end{pmatrix} * v_1^T + \begin{pmatrix} 0 \\ 0 \\ 0 \\ -2 \\ 0 \\ 0 \end{pmatrix} * v_2^T + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -2 \end{pmatrix} * v_3^T \tag{3.31}$$

$$\text{Dérivées par rapport à } m_2 : \begin{cases} \frac{\partial l_1^2}{\partial m_2} = 0 \\ \frac{\partial l_2^2}{\partial m_2} = 2(M_2 - M_0)^T = 2v_2^T \\ \frac{\partial l_3^2}{\partial m_2} = 0 \\ \frac{\partial l_4^2}{\partial m_2} = 2(M_2 - M_1)^T = 2(v_2 - v_1)^T \\ \frac{\partial l_5^2}{\partial m_2} = -2(M_3 - M_2)^T = 2(v_2 - v_3)^T \\ \frac{\partial l_6^2}{\partial m_2} = 0 \end{cases} \tag{3.32}$$

$$Soit : \frac{\partial L}{\partial m_2} = 2 * \begin{pmatrix} 0 \\ v_2^T \\ 0 \\ (v_2 - v_1)^T \\ (v_2 - v_3)^T \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ -2 \\ 0 \\ 0 \end{pmatrix} * v_1^T + \begin{pmatrix} 0 \\ 2 \\ 0 \\ 2 \\ 2 \\ 0 \end{pmatrix} * v_2^T + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -2 \\ 0 \end{pmatrix} * v_3^T \quad (3.33)$$

$$\text{Dérivées par rapport à } m_3 : \begin{cases} \frac{\partial l_1^2}{\partial m_3} = 0 \\ \frac{\partial l_2^2}{\partial m_3} = 0 \\ \frac{\partial l_3^2}{\partial m_3} = 2(M_3 - M_0)^T = 2v_3^T \\ \frac{\partial l_4^2}{\partial m_3} = 0 \\ \frac{\partial l_5^2}{\partial m_3} = 2(M_3 - M_2)^T = 2(v_3 - v_2)^T \\ \frac{\partial l_6^2}{\partial m_3} = -2(M_1 - M_3)^T = 2(v_3 - v_1)^T \end{cases} \quad (3.34)$$

$$Soit : \frac{\partial L}{\partial m_3} = 2 * \begin{pmatrix} 0 \\ 0 \\ v_3^T \\ 0 \\ (v_3 - v_2)^T \\ (v_3 - v_1)^T \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -2 \end{pmatrix} * v_1^T + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -2 \\ 0 \end{pmatrix} * v_2^T + \begin{pmatrix} 0 \\ 0 \\ 2 \\ 0 \\ 2 \\ 2 \end{pmatrix} * v_3^T \quad (3.35)$$

Chacune de ces dérivées partielles est une matrice, c'est une forme linéaire en v. On peut agréger les expressions des dérivées, en utilisant un tenseur d'ordre 3 de dimensions 6x3x4, DLM :

$$DLM = \frac{\partial L}{\partial m} = \left(\frac{\partial L}{\partial m_0}, \frac{\partial L}{\partial m_1}, \frac{\partial L}{\partial m_2}, \frac{\partial L}{\partial m_3} \right) \quad (3.36)$$

On peut écrire 3 matrices DL_{1-3} en agrégeant les termes correspondant aux 4 sommets :

$$DL_1 = DLM(:, 1, :) = \begin{pmatrix} -2 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 2 & -2 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & -2 \end{pmatrix} \quad (3.37)$$

$$DL_2 = DLM(:,2,:) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ -2 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & -2 & 2 & 0 \\ 0 & 0 & 2 & -2 \\ 0 & 0 & 0 & 0 \end{pmatrix} \tag{3.38}$$

$$DL_3 = DLM(:,3,:) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -2 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -2 & 2 \\ 0 & -2 & 0 & 2 \end{pmatrix} \tag{3.39}$$

De telle sorte que le tenseur DLM puisse s'écrire (de façon analogue) :
Forme linéaire qui fait passer de la matrice v (3x3) au tenseur DLM (6x3x4)

$$DLM = \frac{\partial L}{\partial m} = DLM(v_1, v_2, v_3)) = DL_1 * v_1^T + DL_2 * v_2^T + DL_3 * v_3^T \tag{3.40}$$

3.3.2 Gradient de déformation, Tenseurs de dilatation et de déformation

Tenseur de dilatation

Le gradient de déformation F est constant sur un tétraèdre (Approximation P1), la relation liant les vecteurs des arêtes du tétraèdre en cas de déformation est la suivante $\forall i, 1 \leq i \leq 6 : v_i = F.V_i$. donc nous pourrons écrire :

$$l_1^2 = \|v_1\|^2 = v_1^T.v_1 = (V_1^T F^T)(FV_1) = V_1^T(F^T F)V_1 = V_1^T C V_1 \tag{3.41}$$

Comme nous l'avons déjà mentionné, les 6 arêtes du tétraèdre peuvent être écrites en fonction des trois premières arêtes, soit la matrice v de 3.23. Les longueurs des ces arêtes peuvent être écrites en fonction de v donc ces longueurs peuvent être écrites sous formes linéaires de $F^T F$. La matrice $C = F^T F$ résulte linéairement des longueurs au carré des arêtes.

$$C = C(L) = \sum_{i=1}^{6} C_i.l_i^2$$

Tenseur des déformations de Green-Saint Venant

En rappelant la définition de $\boldsymbol{E}$:

$$\boldsymbol{E} = \frac{1}{2}(C - I) \tag{3.42}$$

Nous pouvons écrire :

$$\boldsymbol{E} = \boldsymbol{E}(L) = \frac{1}{2}(\sum_{i=1}^{6} C_i.l_i^2 - I) \tag{3.43}$$

Le tenseur de déformation $\boldsymbol{E}$ étant nul en configuration initiale, nous pouvons écrire :

$$\boldsymbol{E} = \frac{1}{2}(\sum_{i=1}^{6} C_i(l_i^2 - l_{i-0}^2) \tag{3.44}$$

Le carré $\boldsymbol{E}^2$ vaut :

$$\boldsymbol{E}^2 = \frac{1}{4}\sum_{i=1}^{6} C_i^2.l_i^4 + \frac{1}{4}\sum_{i<j}(C_iC_j + C_jC_i).l_i^2 l_j^2 - \frac{1}{2}\sum_{i=1}^{6} C_i.l_i^2 + \frac{1}{4}I \tag{3.45}$$

Invariants $\boldsymbol{E}$: $Tr(\boldsymbol{E})$ et $Tr(\boldsymbol{E}^2)$

On note VTr et MTr respectivement le vecteur des traces des C_i et la matrice des traces de C_iC_j. Avec cette notation, on a :

$$Tr(\boldsymbol{E}) = \frac{1}{2}(\sum_{i=1}^{6} Tr(C_i).l_i^2 - 3) = \frac{1}{2}VTr.L - \frac{3}{2} \tag{3.46}$$

On peut aussi écrire :

$$Tr(\boldsymbol{E}) = \frac{1}{2}\sum_{i=1}^{6} Tr(C_i).(l_i^2 - l_{i-0}^2) = \frac{1}{2}VTr.\Delta L \tag{3.47}$$

et pour $Tr(\boldsymbol{E}^2)$:

$$Tr(\boldsymbol{E}^2) = \frac{1}{4}L^T.MTr.L - \frac{1}{2}VTr.L + \frac{3}{4} \tag{3.48}$$

ou encore :

$$Tr(\boldsymbol{E}^2) = \frac{1}{4}\Delta L^T.MTr.\Delta L - \frac{1}{2}VTr.\Delta L \tag{3.49}$$

Energie interne et forces exercées

Si l'on dispose sur un domaine d'une énergie interne W, on peut déterminer la force interne en un point X par dérivation de l'énergie :

$$F(X) = \frac{\partial W}{\partial X} \tag{3.50}$$

En particulier s'il y a des points particuliers (nœuds d'un maillage), ceci donne pour ces nœuds :

$$F_i = \frac{\partial W}{\partial m_i} \tag{3.51}$$

Pour un tétraèdre T_k en approximation P1, si l'on dispose d'une expression de son énergie interne W, fonction des longueurs L :

$$W_k = W(T_k) = W(L_k) \tag{3.52}$$

On cherche à déterminer la valeur pour chaque tétraèdre T_k, et pour chacun des 4 nœuds du tétraèdre $m_{0..3}$, de l'expression :

$$F_{i,k} = \frac{\partial W_k}{\partial m_i} \tag{3.53}$$

$F_{i,k}$ étant la contribution à la force au nœud i du tétraèdre k. C'est un vecteur de dimension 3.
On peut agréger les 4 vecteurs de forces du tétraèdre k, en une matrice MF_k de dimension 3x4 :

$$MF_k = \begin{pmatrix} \frac{\partial W_k}{\partial m_0} & \frac{\partial W_k}{\partial m_1} & \frac{\partial W_k}{\partial m_2} & \frac{\partial W_k}{\partial m_3} \end{pmatrix} = \frac{\partial W_k}{\partial m} \tag{3.54}$$

W_k étant une fonction de L, nous pouvons écrire :

$$\frac{\partial W_k}{\partial m}(L_k) = \frac{\partial W_k}{\partial L}\frac{\partial L}{\partial m} \tag{3.55}$$

On peut établir une relation pour le deuxième terme qui est un tenseur d'ordre 3 de dimensions 6x3x4 (equation 3.40), Ceci nous permet d'écrire pour MF_k une loi valable pour toute fonction d'énergie W :

$$MF_k = \frac{\partial W_k}{\partial L}.(DL_1 * v_1^T + DL_2 * v_2^T + DL_3 * v_3^T) \tag{3.56}$$

L'énérgie interne d'un matériau de Saint Venant Kirchhoff en fonction de $\boldsymbol{E}$ est selon l'équation 3.57, pour un tétraèdre T_k, élément $P1$ (hypothèse de loi affine sur le tétraèdre), le tenseur des déformations est constant sur le tétraèdre et vaut E_k, et en notant $\|Tk\|$ le volume du tétraèdre en configuration initiale, l'énergie interne du tétraèdre s'écrit :

$$\overline{W}(T_k) = \|T_k\|\left(\frac{1}{2}(\lambda(Tr(\boldsymbol{E}))^2 + 2\mu Tr(\boldsymbol{E}^2)\right) \tag{3.57}$$

En reprenant les expressions de $Tr(\boldsymbol{E})$ de l'équation 3.47 et $Tr(\boldsymbol{E}^2)$ de l'équation 3.49, nous obtenons W en fonction de L

$$W = \frac{1}{4}L^T.\left(\frac{\lambda}{2}(VTr.VTr^T) + \mu.MTr\right)L - \left(\frac{3\lambda}{4} + \frac{\mu}{2}\right).VTr^T.L + \left(\frac{9\lambda}{4} + \frac{3\mu}{4}\right) \quad (3.58)$$

Que l'on peut aussi écrire en fonction de ΔL :

$$W = \frac{1}{4}\Delta L^T.\left(\frac{\lambda}{2}(VTr.VTr^T) + \mu.MTr\right)\Delta L - \frac{\mu}{2}VTr^T.\Delta L \quad (3.59)$$

En notant :

$$MW = \frac{\lambda}{8}(VTr.VTr^T) + \frac{\mu}{4}MTr \qquad \text{matrice de dimension 6x6} \quad (3.60)$$

$$VW1 = -\left(\frac{3\lambda}{4} + \frac{\mu}{2}\right).VTr^T \qquad \text{vecteur ligne de dimension 6} \quad (3.61)$$

$$VW2 = -\frac{\mu}{2}.VTr^T \quad (3.62)$$

On peut écrire :

$$\overline{W} = L^T.MW.L + VW1.L + \left(\frac{9\lambda}{4} + \frac{3\mu}{4}\right) \quad (3.63)$$

Ou :

$$\overline{W} = \Delta L^T.MW.\Delta L + VW2.\Delta L \quad (3.64)$$

MW, $VW1$ et $VW2$ sont une matrice et des vecteurs qui peuvent être pré-calculés à partir de la configuration initiale du tétraèdre P1.
La dérivée de l'énergie s'écrit simplement à partir de la formule 3.64 :

$$\frac{\partial W}{\partial L} = L^T.MW + VW1.L \quad (3.65)$$

Les forces fournies par le tétraèdre sont obtenues à partir de l'équation 3.56, on obtient pour MF_k une fonction linéaire de L et de v :

$$MF_k = ((L^T.MW{+}VW1).DL1).v_1^T{+}(L^T.MW{+}VW1).DL2).v_2^T{+}((L^T.MW{+}VW1).DL3).v_3^T \quad (3.66)$$

MW et $VW1-3$: construit à partir de λ, μ, VTr et MTr
VTr et MTr : construits à partir de B (inverse de V) et de matrices constantes
$DL1-3$: matrices constantes
D'où en fonction de ΔL

$$MF_k = \Delta L^T(MW.DL1).v_1^T + \Delta L^T(MW.DL2).v_2^T + \Delta L^T(MW.DL3).v_3^T \quad (3.67)$$

3.4 Algorithme et implémentation

Nous avons implémenté notre modèle pour un maillage tétraèdrique. Nous commençons par une phase d'initialisation, Il s'agit de stocker les positions initiales puis calculer pour chaque tétraèdre, la matrice B (inverse de V), le volume initial, VTr, MTr, MW, Dl_i ainsi que le produit $MW.Dl_i$
Après cette initialisation nous appliquons l'algorithme suivant : Pour chaque nœud i :

- Initialiser la force à 0 ;

Pour chaque arête j :

- Calculer L et ΔL

Pour chaque Tétraèdre k :

- Calculer v
- Calculer les forces sur les 4 sommets du tétraèdre selon l'équation 3.67 et les cumuler pour chaque

3.5 Validation de la méthode

Dans cette partie, on se propose de tester notre modèle élastique en grandes déformations dans le cas de traction et compression de cube. Il s'agit d'un cube (figure 3.2) de 50 mm de côté (noté L_0) mis en compression ou en traction à vitesse constante de 1mm/s, de manière uniaxiale selon la direction Z. Les directions X et Y sont libres. Ce test nous permet de déterminer analytiquement la contrainte en fonction de la déformation au cours de la compression, afin de les comparer aux résultats numériques.

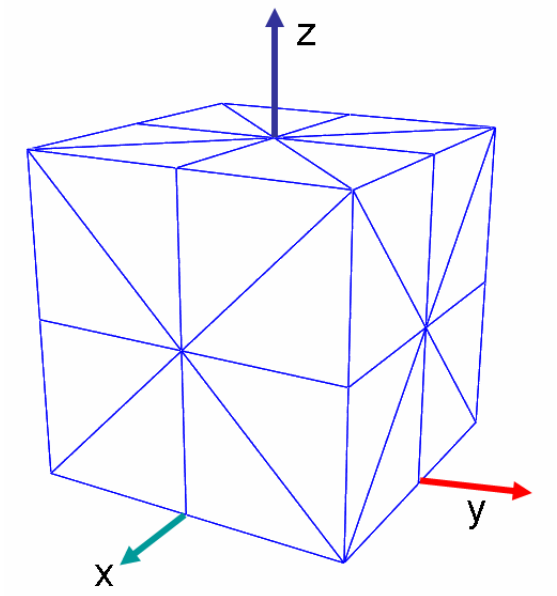

Fig. 3.2 – test de validation

3.5.1 Résultat analytique

Lors de la compression homogène d'un cube, le tenseur des déformations F, s'écrit :

$$F = \begin{pmatrix} F_{xx} & 0 & 0 \\ 0 & F_{yy} & 0 \\ 0 & 0 & F_{zz} \end{pmatrix} \tag{3.68}$$

où F_{xx}, F_{yy} et F_{zz} sont les déformations selon les axes,x, y et z. dans le cas d'une compression homogène selon z on a :

$$F_{xx} = F_{yy} = F_t \tag{3.69}$$

et

$$F_{zz} = \frac{l}{L} = F_l \tag{3.70}$$

Où l est la longueur du côté du cube pendant la compression, F_l pour désigner la déformation longitudinale et F_t pour désigner la déformation tangentielle. On déduit de l'équation 3.68 alors aisément les tenseur de Cauchy (droite et gauche) :

$$C = \begin{pmatrix} F_{xx}^2 & 0 & 0 \\ 0 & F_{yy}^2 & 0 \\ 0 & 0 & F_{zz}^2 \end{pmatrix} \tag{3.71}$$

Le tenseur de Green-Saint Venant $\boldsymbol{E}$ s'écrit selon l'équation 3.18

$$E = \frac{1}{2} \begin{pmatrix} F_t^2 - 1 & 0 & 0 \\ 0 & F_t^2 - 1 & 0 \\ 0 & 0 & F_l^2 - 1 \end{pmatrix} \tag{3.72}$$

L'énérgie interne d'un matériau de Saint Venant Kirchhoff en fonction de $\boldsymbol{E}$ est selon l'équation 3.57, ce qui nous permet de calculer le second tenseur de Piola-Kirchhoff S qui s'obtient en dérivant l'énergie de déformation W par rapport à C (équation 3.15) Donc,

$$S = \frac{\lambda}{2}\Big((F_l^2 - 1) + 2(F_t^2 - 1)\Big)I + \mu \begin{pmatrix} F_t^2 - 1 & 0 & 0 \\ 0 & F_t^2 - 1 & 0 \\ 0 & 0 & F_l^2 - 1 \end{pmatrix} \tag{3.73}$$

Or dans le cas d'une compression uniaxiale selon z et dans le cas où les bords sont libres, on a :

$$S_{xx} = S_{yy} = 0 \tag{3.74}$$

Ce qui donne alors :

$$S_{zz} = \mu \frac{3\lambda + 2\mu}{2(\lambda + \mu)}(F_l^2 - 1) \tag{3.75}$$

d'ou nous concluons P_{11} du premier tenseur de Piola-Kirchhoff d'après l'équation 3.13

$$P_{zz} = F_l S_{zz} = \mu \frac{3\lambda + 2\mu}{2(\lambda + \mu)} F_l (F_l^2 - 1) \tag{3.76}$$

La force résultante f_S suivant z sur la surface de cube est égale à la contrainte suivant z, σ_{zz} multipliée par la surface déformée. Or, cette surface déformée est égale à la surface initiale S_0 multipliée par F_t^2, d'où :

$$f_S = \sigma_{zz} S_0 F_t^2 \tag{3.77}$$

Or, selon l'équation 3.14 nous pouvons écrire :

$$\sigma_{zz} = \frac{F_l}{F_t^2} S_{zz} \tag{3.78}$$

D'où la force :

$$f_S = P_{zz} S_0 \tag{3.79}$$

3.5.2 Comparaison analytique-numérique

Pour un module d'Young de 0.02 MPa, qui est l'ordre de grandeur du module d'Young des tissus mous [MAA99], on s'intéresse aux résultats de compression du cube. Dans un premier temps, nous validons notre méthode sur un cube unitaire, c'est à dire la taille de maille correspond au côté du cube, sinon le systeme ne se stabilise pas, on retrouve bien le phénomène de retour élastique puisque notre comportement élastique ne dissipe pas encore l'énergie. La comparaison des courbes contrainte-déformation analytique et numérique donne le résultat suivant (Figure 3.3). Nous remarquons que notre courbe numérique correspond parfaitement à la courbe analytique dans le cas d'un cube unitaire. Prenons le cas du cube décrit ci-dessus (figure 3.2). La comparaison des courbes contrainte-déformation analytique et numérique donne le résultat suivant (Figure 3.4). La courbe numérique correspond à la courbe analytique dans le cas d'un cube maillé. Mais nous constatons deux anomalies :

1. Les oscillations dûes à l'élasticité du modèle d'où la nécessité d'un terme dissipatif. nous traitons cette anomalie dans le paragraphe suivant.
2. Comme il s'agit d'une validation de déformation en grande déformation, il s'agit dans notre cas d'une compression de 100%, il arrive que des tétraèdres seront inversés ce que nous appelons flambage numérique car cette anomalie n'existe pas en cas réel le falmbage dans le cas expérimental se manifeste par une déchirure des tissus. Pour remedier à ce problème soit nous éliminons le tétraèdre qui a subit une grande déformation ce qui simule la déchirure, soit nous ajoutons un terme de pénalité de volume sur les forces calulées. Notons que cette dernière méthode affecte les résultats numériques.

Ajoutons que cette deuxième anomalie est moins présente en cas de traction comme le montre la figure 3.5 Ces essais nous permettent de mettre en évidence que des instabilités numériques apparaissent surtout dans le cas de la compression. Une approche possible pour atténuer ou supprimer les instabilités numériques consiste à ajouter une composante visqueuse dans le comportement du matériau.

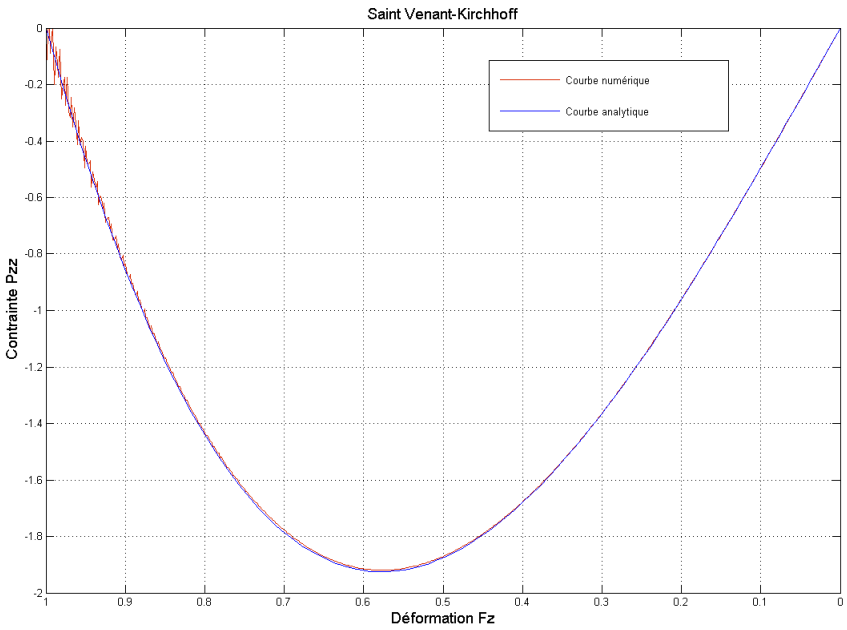

FIG. 3.3 – Comparaison courbes contrainte-déformation analytique et numérique (cube unitaire)

3.6 Le modèle Visco-élastique

Comme nous venons de le mentionner dans le paragraphe précédent, nous proposons un nouveau modèle prenant en compte, en plus du comportement hyperélastique, un régime visqueux.

3.6.1 Différents modèles de Viscoélasticité

Il existe différents modèles de viscoélasticité [WAR93]. Nous nous sommes intéressés aux modèles les plus simples :

- Le modèle de Maxwell (Figure 3.6) est composé d'un ressort de Module d'Young E et d'un amortisseur de viscosité ν en série. La déformation totale est la somme des déformations dues au comportement élastique (ε_e) et visqueux (ε_ν) :$\varepsilon = \varepsilon_e + \varepsilon_\nu$.
- Le modèle de Kelvin-Voigt (Figure 3.7) est composé d'un ressort de Module d'Young E et d'un amortisseur de viscosité ν en parallèle. Dans ce cas, la déformation est la même, $\varepsilon_e = \varepsilon_\nu$. La contrainte s'exprime sous la forme : $\sigma = \sigma_e + \sigma_\nu$

L'étude d'un modèle viscoélastique a été conduite afin de modéliser un comportement élastique tout en atténuant les instabilités numériques. En d'autres termes, on cherche à caractériser un comportement viscoélastique pour lequel la partie visqueuse doit permettre d'éliminer les instabilités numériques, mais doit rester assez faible pour ne pas

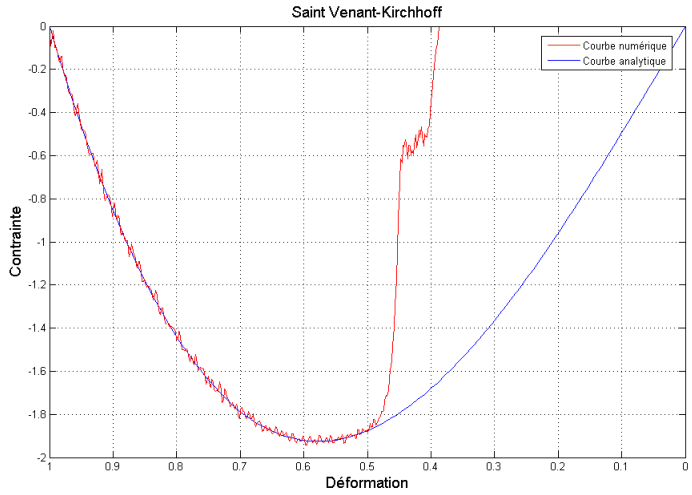

FIG. 3.4 – Comparaison courbes contrainte-déformation analytique et numérique

perturber le comportement purement élastique.
Dans le cas d'un comportement élastique, la loi de comportement relie le tenseur des contraintes à celui des déformations sous la forme générale :

$$\sigma = f(\varepsilon)$$

Dans le cas du modèle de Maxwell, nous aurons donc :

$$\sigma = f(\varepsilon_e + \varepsilon_\nu)$$

Si l'on cherche à avoir un comportement total proche de celui du comportement purement élastique, ε_ν doit être négligeable devant ε_e, ce qui revient à écrire :

$$\sigma = f(\varepsilon_e)$$

On retombe donc sur le même système que précédemment (lors des instabilités numériques). Dans le cas du modèle de Kelvin-Voigt, la loi de comportement se met sous la forme générale :

$$\sigma = \sigma_e + \sigma_\nu$$

Il est possible de trouver dans ce cas, une contrainte due au comportement visqueux assez faible pour ne pas perturber le comportement total mais suffisante pour diminuer les instabilités numériques.
C'est pour cela que nous avons choisi le modèle de Kelvin-Voigt.

3.6.2 Formulation Viscoélastique

Comme nous avons choisi le modèle de Kelvin-Voigt, le système à résoudre restera le même que celui que nous venons de décrire tout en ajoutant le terme dû à la viscosité.

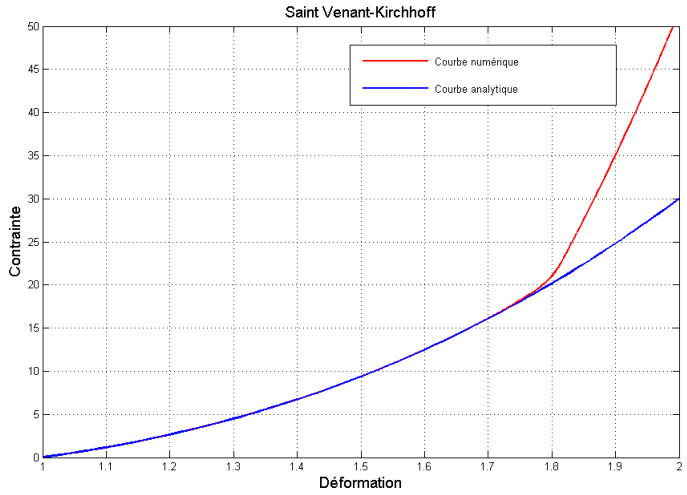

FIG. 3.5 – Comparaison courbes contrainte-déformation analytique et numérique en cas de traction

Notre problème se résume à ajouter lors du calcul des forces sur chaque nœud un terme de viscosité venant de toutes les liaisons assosiées à ce nœud. Il s'agit d'une formulation d'un terme d'amortissement s'associant à la différence de vitesse de chaque nœud avec celle du centre de gravité du tétraèdre associé et proportionnel au volume initial du tétraèdre. La figure 3.8 nous montre clairement l'intérêt du terme dissipatif d'un coefficient de 10^{-3}. Comme nous l'avons déjà mentionné la force due au comportement visqueux doit être assez faible pour ne pas perturber le comportement total mais suffisante pour diminuer les instabilités numériques. La figure 3.9 nous montre l'erreur qu'a introduit un grand coefficient d'amortissement. Nous avons aussi testé notre modèle sur plusieurs tailles de maillages, la figure 3.10 nous montre le résultat.

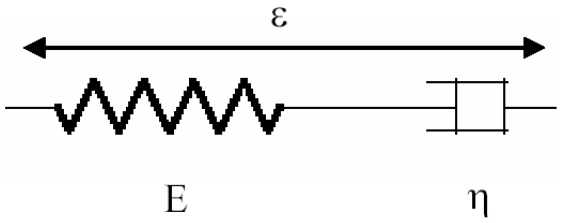

FIG. 3.6 – Le modèle de Maxwell

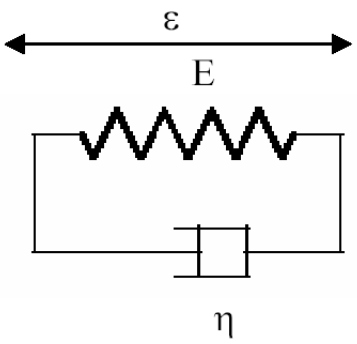

FIG. 3.7 – Le modèle de Kelvin-Voit

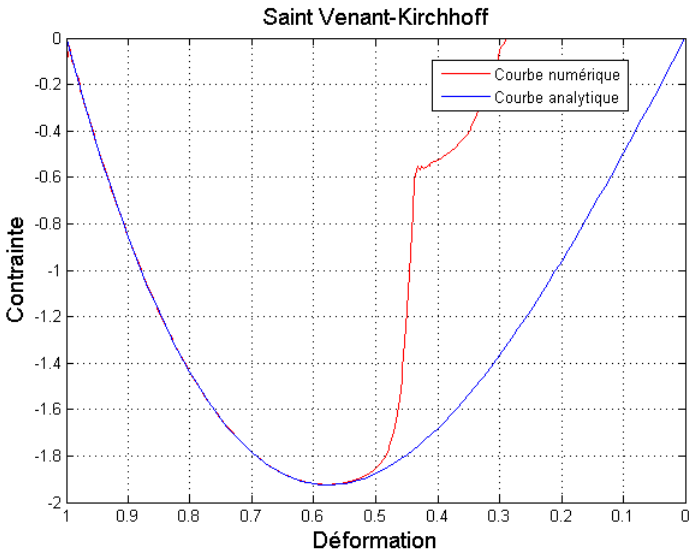

FIG. 3.8 – Comparaison courbes contrainte-déformation analytique et numérique en comportement viscoélastique

3.7 Conclusion

Nous avons présenté dans ce chapitre notre approche pour la modélisation temps réel des corps déformables. Il s'agit d'une méthode physique, qui offre la possibilité de paramétrer le modèle à partir de mesures expérimentales, et de construire une solution générique applicable à une grande variété de matériaux. Il s'agit d'une approche issue de la Mécanique des Milieux Continus ; c'est une synthèse entre deux presentations répandues : les masses-ressorts et les masses tenseurs. Elle se base sur le modèle des matériaux de Saint Venant Kirchhoff qui correspondent à une classe de matériaux hyperélastiques. Notre contribution principale concerne la modélisation du comportement viscoélastique d'après une formulation simple qui offre à notre algorithme une complexité linéaire par rapport au nombre de tétraèdre. Ajoutons que notre système présente un avantage majeur concernant le temps de calcul. Cet algorithme est linéaire par rapport au nombre des tétraèdres. Nous avons calculé le nombre d'opération pour notre modèle et celui du modèles masse-tenseur. Notre modèle peut aller jusqu'à 100 fois plus vite pour un organe de plus que 10 000 tétraèdres. Nous reprenons le tableau récapitulatif 3.1 en introduisant notre modèle [GC06].

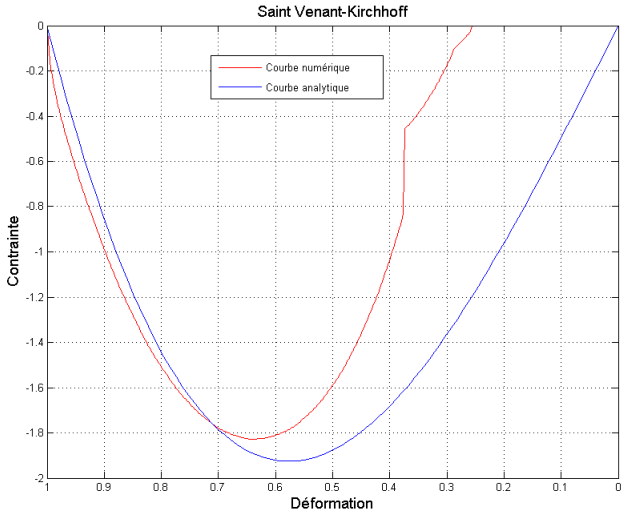

FIG. 3.9 – Erreur entre les courbes contrainte-déformation analytique et numérique en comportement viscoélastique suite à un grand coefficient d'amortissement

TAB. 3.1 – Tableau récapitulatif des avantages et inconvénients des différentes méthodes y inclus notre modèle

	M-R	**E-F**	**M-T**	MLE
Facilité de mise en œuvre	⋆⋆⋆	⋆	⋆⋆	⋆⋆
« Temps réel »	⋆⋆⋆	⋆	⋆⋆	⋆⋆⋆
Réalisme physique	⋆	⋆⋆⋆	⋆⋆	⋆⋆
Changement de topologie	⋆⋆⋆	⋆	⋆⋆⋆	⋆⋆⋆
Grandes déformations	⋆⋆	⋆⋆	⋆	⋆⋆
Grands déplacements	⋆⋆⋆	⋆	⋆	⋆⋆⋆

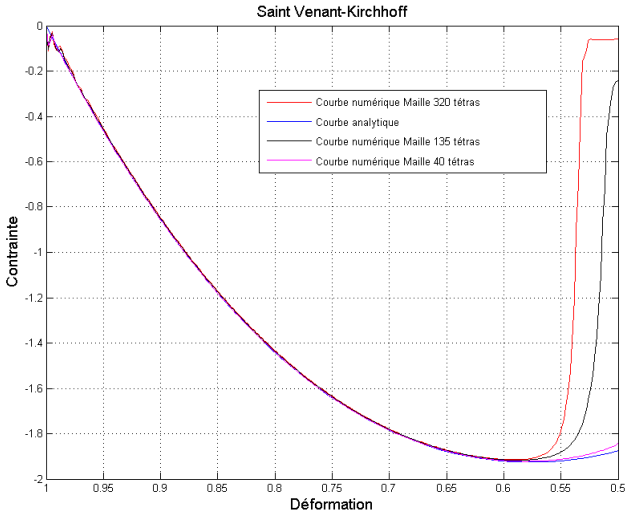

FIG. 3.10 – Courbes contrainte-déformation analytique et numérique pour plusieurs tailles de maillage

Chapitre 4

Interaction avec les organes modélisés

Après le développement des modèles de corps déformables, nous nous intéressons à l'interaction avec les organes virtuels. Il s'agit des trois problématiques principales : la détection des collisions ou des auto-collisions, la gestions de ces collisions et aussi la découpe.

4.1 Introduction

Après le développement des modèles de corps déformables, nous nous intéressons à l'interaction avec les organes virtuels. Il s'agit des trois problématiques principales : la détection des collisions ou des auto-collisions, la gestions de ces collisions et aussi la découpe. Notons que ces travaux ont été accomplis dans le cadre de deux stages de DEA. Avec Antoine ARNAIL nous avons travaillé sur les collisions et auto-collisions [Arn05] alors qu'avec Xavier BRUN nous avons traité la découpe [Br04]. Ces deux travaux ont été implémentés avec le modèle masse-ressort.

4.2 Détection des collisions

Dans la simulation de chirurgie endoscopique, l'utilisation du simulateur fait évoluer la position des instruments (pinces, crochets, résectoscopes,...) dans une scène graphique 3D et les fait interagir avec les organes. C'est le cas de l'interaction outil-organe d'où la détection des collisions. L'organe qui est un objet déformable et découpable peut entrer dans une configuration physiquement impossible où deux de ses parties se superposent. C'est le cas des auto-collisions. Cette interaction peut se diviser en deux problèmes distincts : la localisation qui consiste à détecter la collision et à la localiser sur les surfaces, le traitement qui est la gestion des collisions détectées pour les éviter ou les gérer avant l'affichage de l'organe.

Comme nous venons de le voir, l'interaction avec les organes déformables ne se limite pas au simple contact entre l'outil de chirurgie et l'organe virtuel. Elle inclut également l'interaction des organes déformables entre eux ou même l'interaction entre différentes parties d'un même organe, c'est le cas des organes qui peuvent se plier sur soi tel un intestin ou un organe qui a subi une découpe. C'est le cas des auto-collisions. Dans la suite, l'utilisation du terme collision désigne la collision outil-organe et l'auto-collision (collision organe-organe).

Les méthodes de détection des (auto-)collisions dépendent fortement du modèle sur lequel on travaille. On trouve deux excellents résumés de ces différentes méthodes de détection publiés respectivement par M.Lin [Lin98] et P.Jiménez [Jim01].

La détection de collision consiste à calculer la distance entre deux objets. Prenons le cas le plus simple dans une simulation. Il s'agit de la détection de collision entre un point et un objet. Ce cas est souvent rencontré dans les simulations munies d'une interface de retour d'effort à 3 degrés de liberté comme le Phantom Desktop que nous avons à disposition. Nous utilisons un maillage en triangles de la surface de l'organe. Ce choix est justifié par le choix du modèle de l'organe déformable que nous avons décrit dans le chapitre précédent. La condition de collision se produit lorsque le point traverse la surface pour passer à l'intérieur de l'objet. Afin de détecter ce passage, on fait un test d'intersection entre le segment qui lie la position actuelle et la position précédente du point et les triangles de la surface(figure 4.1).

Si le segment traverse la surface de l'objet, on cherche une projection sur les triangles de la surface afin de déterminer la profondeur de pénétration. Notons que nous travaillons

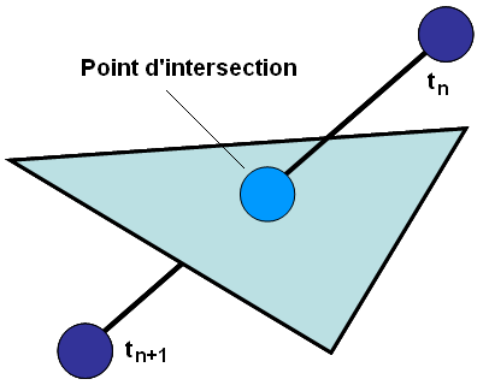

FIG. 4.1 – *Détection de collision point-objet*

avec des organes déformables ce qui signifie que les optimisations basées sur la géométrie des objets et les pré-calculs sont inutilisables. Ces optimisations doivent s'adapter aux déformations c'est à dire être reconstruites ou modifiées à chaque pas d'échantillonnage. La détection de collision dépend donc du nombre de triangles de surface de l'organe déformable. Ajoutons que nous ne pouvons pas limiter la représentation de l'outil de chirurgie à un simple point d'interaction. Dans la suite nous traitons l'outil comme un organe à maillage de surface triangulaire. Nous pouvons donc adopter les mêmes méthodes pour la détection des collisions et la détection des auto-collisions.

Donc, la rapidité d'un algorithme de détection des collisions est affectée par les conditions suivantes :

1. La taille des objets virtuels (nombre de triangles de surface). Plus grand ou plus complexe est un modèle, plus grand est le nombre des tests de collision à faire.
2. Le mouvement d'un objet dans l'environnement virtuel, sa déformation et le changement de sa forme surtout lors de la découpe qui entraîne un changement du nombre des triangles de surface.
3. Le nombre de points de contact simultanés. Cette condition influe surtout sur la gestion des collisions.

Nous nous attarderons donc plus spécifiquement sur les méthodes de détection de collision entre triangles, en postulant une cohérence temporelle dans l'évolution des triangles. Il s'agit de la detection de contact de type triangle-triangle.

Les contraintes dont il faut tenir compte :

– Les triangles sont déformables
– Le nombre de triangles peut évoluer
– Pour le même organe, il existe déjà un lien entre triangles voisins : ils sont attachés.

La difficulté est le nombre de tests (en $\bigcirc(n^2)$) où n est le nombre de triangles. Pour améliorer le temps de calcul, il faut commencer par éliminer une partie des paires de triangles du test. Nous distinguons trois cas de détection de collision :

– Collision d'un objet avec une primitive ou un objet.

– Collision de N objets.
– Autocollision.

Pour étudier le troisième cas qui est le plus difficile en termes de complexité nous allons présenter les différents algorithmes proposés pour résoudre les problèmes à N corps, nous discuterons ensuite de leurs applications dans le cas des auto-collisions.

La plupart du temps, les algorithmes sont séparés en deux catégories principales :

La détection de collision exacte. Il s'agit de tests d'interférence entre les primitives géométriques de base de l'environnement virtuel comme nous venons de le voir dans le cas du contact objet-point. Vu que les algorithmes de détection de collision doivent être très rapides pour assurer la stabilité du retour d'effort, les premières simulations à retour d'effort ne géraient que les interactions entre un objet et un point représentant en général l'extrémité de l'outil. Ces dernières années, l'augmentation de la puissance des ordinateurs a permis le développement des algorithmes qui gèrent les collisions des objet rigides ainsi que les collisions qui concernent les modèles déformables comme celles que nous utilisons pour la simulation chirurgicale.

La détection de collision approximative ou grossière. Il s'agit d'une recherche pour la localisation des régions potentielles de contact, surtout pour les grands modèles. Il n'est pas envisageable de tester toutes les paires possibles de primitives qui forment leurs surfaces (des triangles dans notre cas). Ces méthodes tendent à limiter le nombre de tests totaux en donnant une idée grossière de la région de l'objet sur laquelle une collision peut exister.

Ces deux catégories de détection de collisions sont complémentaires pour une optimisations de la détection. La localisation des régions potentielles vient en première phase. Elle est suivie par la détection de collision exacte. Toute l'optimisation de la détection de collision se base sur les méthodes rapides de localisation des régions potentielles de contact.

4.2.1 Méthodes de détection de collision approximative

Il existe trois familles de méthodes pour la détection de collision approximative.

a) Les méthodes de tri

b) les méthodes de décomposition ou de partition spatiale

c) les méthodes hiérarchiques

Nous traitons ces trois catégories tout en prenant en compte la cohérence temporelle. C'est l'hypothèse que les positions relatives des différentes primitives changent peu entre deux instants de temps t_n et t_{n+1}. Cela veut dire que si à l'instant t_n il y a une collision entre les organes, la région de contact est de nouveau la région la plus probable de contact à t_{n+1}, et inversement si à l'instant t_n, les primitives sont très distantes, il est moins probable d'entrer en contact à t_{n+1}.

Ajoutons qu'une grande partie des algorithmes fait appel à la technique des boîtes englobantes afin d'éviter des tests de collision pour tous les triangles de la surface d'un

organe. Il s'agit d'utiliser des structures simples qui englobent chaque triangle et parfois un ensemble de triangles.

a) Méthodes de tri

Ordonnancement On fixe une borne supérieure V_{max} à la vitesse de déplacement des différentes primitives (cohérence temporelle). On suppose de plus que le pas d'intégration τ est fixé. Deux objets T_i et T_j distants de $d(i,j)$ n'entreront pas en collision durant $n_{i,j}$ étapes si

$$n_{i,j} = \lfloor \frac{d(i,j)}{V_{max} * \tau} \rfloor.$$

L'idée est alors d'ordonnancer [kim97] la recherche des collisions selon ce temps $n_{i,j}$

Si $V_{max} * \tau$ est petit devant la dimension du modèle, $n_{i,j}$ sera non nul pour un certain nombre de paires donc il est inutile de tester si la paire (T_i, T_j) entre en collision durant $n_{i,j}$ étapes.

En pratique les "distances" sont calculées une première fois au démarrage de l'algorithme pour chaque paire, puis insérées dans une liste triée selon les $t_{i,j}$.

Avantages dans le cas d'un modèle déformable pour interface haptique :

- L'interface haptique impose une constante de temps τ petite. Si la vitesse maximum V_{max} est raisonnable on peut espérer faire très peu de tests à chaque étape.
- Le fait de travailler sur des paires permet d'éliminer les cas dégénérés (triangles voisins) contrairement aux autres algorithmes qui feront toujours un test dans ce cas.
- Flexibilité : On ne passe pas nécessairement par des boîtes englobantes.

Limites dans le cas d'un modèle déformable pour interface haptique :

- Coût mémoire
- Le nombre de calculs effectués à chaque étape est probablement très variable d'une étape à l'autre.

Sweep and Prune On calcule dans un premier temps les boîtes englobantes orientées sur les axes (*Axis Aligned Bounding Box* / AABB), la complexité de cette étape est en $\bigcirc(n)$.

On trie ensuite ces boîtes selon les trois axes, ce qui est réalisable en $\bigcirc(nlog(n))$. Cependant la complexité espérée de cet algorithme de tri est de $\bigcirc(n + e_x + e_y + e_z)$ si la liste est correctement triée (cohérence temporelle/ e_α est le nombre d'échanges sur $O\alpha$). Si le nombre d'échanges est faible (ou lorsque la liste subit peu de modifications) on arrive à $O(n)$ en utilisant le tri par insertion.

Il faut réaliser le tri 3 fois $((x_{min}, x_{max}), (y_{min}, y_{max}), (z_{min}, z_{max}))$, il est aisé de mettre à jour une liste de paires en collisions lors de ce tri, cela n'augmente pas la complexité.

En pratique sweep and prune [Smi95] peut être utilisé au sein d'une méthode hiérarchique pour traiter un ensemble de boîtes englobantes.

Avantages dans le cas d'un modèle déformable pour interface haptique :

- Complexité espérée faible

- Coût mémoire faible
- Mise à jour facile
- Deux voisins entrent systématiquement en collision : on peut éliminer définitivement les paires de voisins au départ.

Limites dans le cas d'un modèle déformable pour interface haptique :
- Complexité théorique en $\bigcirc(n^2)$

b) Partition spatiale

Avec les algorithmes de cette famille, on décompose l'espace virtuel ou une boîte englobante en boites parallélipédiques. Avec des tests algébriques simples, nous pouvons vérifier rapidement si deux ou plusieurs primitives se trouvent simultanément dans la même boite.

Grille d'occupation On discrétise l'espace dans une grille d'occupation. Le placement se fait en $\bigcirc(n)$ si la grille est largement plus grande que les triangles. En pratique ce sera toujours le cas vu que le but est de ne tester que les intersections concernant chaque point de la grille. Il faut donc parcourir ces N^3_{grille} points.

Le choix de la taille de la grille doit en conséquence vérifier $N^3_{grille} \ll n^2$.

Avantages dans le cas d'un modèle déformable pour interface haptique :
- Très facile à actualiser
- La table de hachage permet de contourner les limites imposées par une partition spatiale simple en offrant une recherche plus rapide des objets en collision. Elle simplifie l'actualisation du modèle ainsi que la recherche d'éléments en collision.

Limites dans le cas d'un modèle déformable pour interface haptique :
- La grille est peu adéquate pour représenter une surface, de nombreux carrés sont vides au centre. La répartition est donc loin d'être uniforme au sein de la grille. On a donc beaucoup de triangles dans une petite portion des éléments de la grille et pour chacun de ces éléments, il faut considérer toutes les paires qu'il contient.

Subspace tree La méthode de classement dans une grille peut être améliorée si l'on utilise un arbre équilibré de recherche des cellules occupées. En utilisant des sphères de tailles constantes comme boîtes englobantes, l'arbre est créé en $\bigcirc(log(n))$ et mis à jour en $\bigcirc(n)$.

c) Méthodes Hiérarchiques

Binary and Multi Space Partitionning Ces méthodes fournissent un arbre triant les objets selon leur position relative, et ce respectivement dans un arbre binaire ou un arbre n-aire. Il s'agit d'une amélioration des méthodes de partition spatiale. Elles sont utilisées dans les jeux pour détecter les collisions mais elles sont peu adaptées au cas des objets déformables.

Avantages dans le cas d'un modèle déformable pour interface haptique :
- Méthode qui peut tirer parti de boîtes englobantes orientées

Limites dans le cas d'un modèle déformable pour interface haptique :
- Les boîtes englobantes orientées apportent peu de gain par rapport à un test directement sur le triangle.
- La mise à jour de l'arbre est trop coûteuse.
- Les triangles d'une surface ne forment pas une partition efficace
- Le choix des nœuds est soit lent soit inefficace

Octree Un Octree est une structure hiérarchique pour représenter ou grouper des objets dans l'espace(figure 4.2. Chaque nœud contient 8 éléments qui sont eux même soit

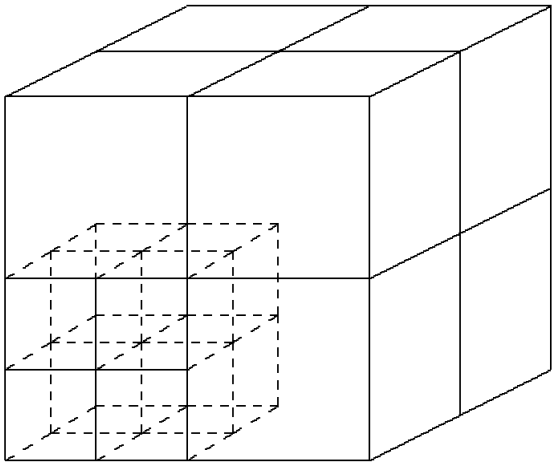

FIG. 4.2 – *octree*

un nœud soit une feuille. Chacun de ces éléments est de dimension identique. La racine est la boîte englobante de tous les éléments, l'arbre est construit récursivement afin d'obtenir un certain nombre d'éléments par feuille. Chaque feuille contient alors une liste des AABB qui ont une intersection non nulle avec l'arbre[Smi95].

On distingue deux types d'octrees, les octrees complets dont toutes les feuilles sont à la même profondeur présentent peu d'intérêt car cela revient à considérer une grille d'occupation, et dans ce cas la structure d'arbre ne nous apporte rien. Par contre si l'octree n'est pas complet, il est nécessaire de choisir un critère d'arrêt en définissant une feuille. Par exemple on peut considérer un arbre de profondeur l avec au plus k éléments par feuille de profondeur inférieure.

Variante : Il existe une variante des octrees où les plans de séparation ne sont pas uniformément répartis, mais calculés de façon à séparer uniformément les triangles (calcul du median en x, y et z).

Avantages dans le cas d'un modèle déformable pour interface haptique :
- On hiérarchise la recherche des paires pouvant avoir une intersection[Lar01].
- L'approche semble plus adaptée que sweep and prune dans le cas où l'on a plusieurs objets, on construit dans ce cas un octree pour chacun d'entre eux [Smi95].

– La mise à jour de l'arbre est simplifiée losrque l'on travaille sur des surfaces paramétriques subissant de faibles déformations [Ryn02].

Limites dans le cas d'un modèle déformable pour interface haptique :

– La mise à jour de l'arbre est coûteuse.

Hierarchical sweep and prune *Méthode de partition utilisant les liens logiques reliant les triangles à trier* : On peut améliorer l'algorithme sweep and prune par une approche multi-échelle [Mad95] qui tient compte de la forme des objets. Cet algorithme a éte implémenté dans le cas d'objets indéformables dont les boîtes englobantes sont triées puis testées à différents niveaux de résolution. Nous pensons que cette approche est généralisable dans le cas où les objets sont déformables.

Dans le cas d'un modèle déformable, il faudra tenir compte du fait que deux objets sont attachés ou non afin de les séparer dans des sous-arbres différents. On peut initialiser l'arbre en étiquetant les composantes connexes dans un premier temps, puis en choisissant des sous ensembles des composantes connexes, par exemple en utilisant une distance de Dijkstra. Une autre possibilité est d'initialiser l'arbre en utilisant une partition de l'espace si l'on est sûr que les objets sont convexes. Cette méthode est plus rapide à initialiser mais moins générale.

Avantages dans le cas d'un modèle déformable pour interface haptique :

– On a les avantages de l'approche hiérarchique et de l'algorithme sweep and prune.

Limites dans le cas d'un modèle déformable pour interface haptique :

– Il faut mettre à jour l'arbre lors de la découpe (étiquetage des composantes connexes)
– Long à implémenter.

Limites des méthodes hiérarchiques A première vue, l'arbre doit être entièrement parcouru dans les cas des autocollisions. Ce parcours est en $\bigcirc(nlog(n))$.

Pour en tirer avantage, il faut donc obtenir des boîtes englobantes plus précises que des cercles ou des carrés afin que le nombre de résultats soit probant. Les OOBBs (*Object Oriented Bounding Box*) par exemple donneront de meilleurs résultats (moins de collisions d'OOBB).

Il est possible de ne parcourir que certaines branches de l'arbre lorsqu'une information sur l'orientation des triangles est disponible. En effet, seuls les côtés extérieurs des triangles peuvent entrer en contact. Pour cela il est possible d'échantilloner un certain nombre de directions comme le propose Rynson et al.[Ryn02], on affecte alors à chaque feuille de l'arbre un masque de d bits, d étant le nombre de directions sur lesquelles se fait la discrétisation. Les directions ayant un produit scalaire positif avec la normale orientée vers l'extérieur de la surface sont à 1, les autres à zéro. Le masque d'un nœud est alors le & logique entre les masques de ses feuilles. Un masque non nul signifie que toutes les normales des triangles ont un produit scalaire non nul deux à deux. Si la surface est continue (les triangles sont connexes), cela implique que les triangles ne s'intersectent pas.

Tableau récapitulatif

Méthode	Complexité	Remarques
Ordonnancement	$\bigcirc(n^2)$	Coût en mémoire important Adapté aux espace peu denses
Sweep and Prune	$\bigcirc(n^2)$	Complexité espérée en $\bigcirc(n + e_x + e_y + e_z)$ e_α étant le nombre d'inversions sur l'axe α
Partitions spatiales	$\bigcirc\,(nlog(n))$	
Méthodes hiérarchiques	$\bigcirc\,(nlog(n))$	Dans le cas des auto-collisions un arbre est inefficace car on doit le parcourir en entier à chaque fois

4.2.2 Méthode choisie

Nous avons choisi d'implémenter l'algorithme sweep and prune car sa complexité espérée est la plus faible. Cela signifie principalement que le nombres de triangles du modèle pourra être étendu proportionnellement à la vitesse de calcul de la machine. Parmi les différents algorithmes cités ci-dessus, d'autres auraient pu être choisis ou testés, mais ils affichent tous une complexité espérée plus importante.

L'hypothèse initiale de cohérence temporelle s'impose presque d'elle même. Étant donné les temps de calcul recherchés, de l'ordre d'une milliseconde, il est cohérent de supposer que les déplacements seront minimes entre deux intégrations.

D'autres critères auraient pu nous conduire à faire des choix différents, notamment la recherche d'un algorithme dont le temps de calcul ne varie pas dans le temps. Aucun des algorithmes ci-dessus ne présente cette propriété, et le temps de calcul dépend toujours du nombre de collisions ou de la répartition des triangles. Sweep and prune ne fait pas beaucoup mieux car son temps de calcul dépend du nombre de triangles (d'AABB en fait) dont les coordonnées sont échangées par l'agorithme de tri. Choisir un temps de calcul constant revient en fait à prendre une méthode qui testera toutes le paires deux à deux. Mieux vaut un temps borné plus faible que ce temps constant.

Choix d'implémentation

Notre première implémentation permet la detection des extra et intra-collisions dans le modèle masse-ressort, en utilisant deux threads séparés pour l'intégration mécanique et la détection. La première limite de cet algorithme est qu'il ne s'effectue pas en temps constant dans le cas général. La première passe prendre les éléments du modèle dans le désordre et les classe selon X, Y et Z. Comme la méthode de tri (tri par insertion) n'est efficace que si les éléments à trier sont quasi-ordonnés, cette première passe est très lente. Il vaut donc mieux la lancer avant de commencer à déplacer les différents éléments, et donc avant tout calcul mécanique. La seconde limite est que cet algorithme est ralenti par des déplacements relatifs importants car on s'éloigne de l'hypothèse initiale de cohérence temporelle.

Une seconde implémentation ne séparant plus, dans deux threads différents, les étapes d'intégration et de détection des collisions se montre finalement plus efficace. En effet la détection ne prend pas de retard sur le modèle mécanique.

Résultats

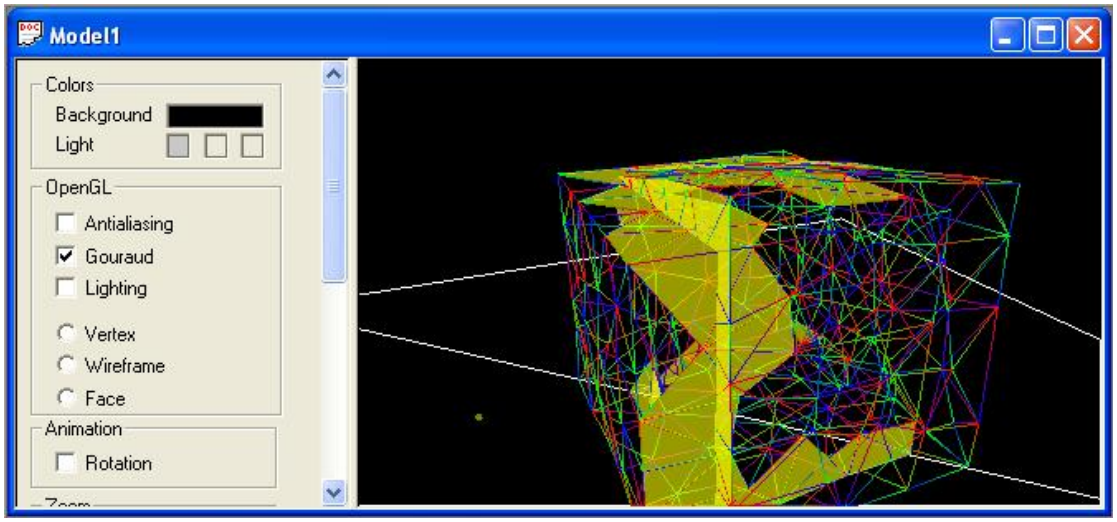

Fig. 4.3 – Détection des collisions

La détection étant effectuée sur des boîtes englobantes orientées sur les axes, on détecte plus de collisions qu'il n'y en a réellement. Ceci est dû au fait que les boîtes englobantes sont plus larges que les objets. Deux boîtes s'interpénètrant est une condition nécessaire mais non suffisante pour que les deux triangles correspondants soient en collision.

On peut affiner la détection en faisant un test supplémentaire[Dev02]. Cependant il peut être utile de détecter une collision avant qu'elle n'arrive, par exemple pour appliquer une pénalité empêchant la collision. Ce test supplémentaire n'est donc pas fait systématiquement.

Lorsque les collisions sont gérées et donc évitées, l'algorithme se montre plus performant car l'algorithme de tri a moins d'inversions à gérer. Ce phénomène renforce la cohérence temporelle car il nous rapproche de l'hypothèse justifiant notre choix : l'ordre des limites de nos boîtes englobantes est peu modifié entre deux étapes de l'algorithme de tri.

Temps de détection et temps réel

Un affichage fluide requiert une fréquence de 30 Hz et donc d'effectuer l'ensemble des calculs toutes les 33 ms. Cependant le retour tactile impose des raffraichssements entre 700 et 1000 Hz. Une troisième limite est le pas d'intégration, de 1 ms pour des raisons de stabilité si la vitesse est bornée dans le modèle, 0.1 ms sinon.

Un affichage temps réel nous impose un calcul plus rapide que ces trois temps de référence. L'ordre de grandeur est donc la milliseconde. Comme les temps de calcul dépendent de la machine utilisée et du nombre de triangles dans le modèle, le fait d'at-

teindre un temps donné n'est pas en soi significatif. Le cube montré en deux exemplaires dans la figure 4.6, est constitué de 353 tétraèdres en interne et 288 triangles en surface. Lorsque les collisions sont gérées, il est affiché seul en moins d'une milliseconde, par deux en moins de 2.5 millisecondes (au minimum 400 fps), par quatre en moins de 6 ms (au minimum 150 fps). Ces résultats sont corrects, mais montrent que l'algorithme n'est pas linéaire.

Nous avons aussi cherché à calculer le temps de calcul de cet algorithme indépendamment des autres opérations du programme. Cependant, après les dernières optimisations implémentées, ce temps devient inférieur à la précision de notre timer qui est de trois millisecondes.

4.3 Gestion des collisions

Le modèle ne doit pas afficher deux surfaces qui s'interpénètrent. Une fois détectée la collision doit donc être évitée ou gérée avant l'affichage du maillage. La méthode de gestion de collision doit être choisie selon plusieurs critères qui sont le temps de calcul, le réalisme du rendu et la stabilité du modèle.

Afin de rester cohérent avec le modèle masse-ressort ou le modèle masse-liaison étendu utilisé pour modéliser le tissu mou, une résolution locale semble plus adaptée.

Nous avons testé les deux méthodes les plus répandues, et nous proposons une approche rarement utilisée basée uniquement sur des contraintes géométriques.

4.3.1 Méthodes classiques de gestion des collisions

Le plus simple est d'éviter les collisions en imposant une force de répulsion entre les objets proches, c'est la méthode des pénalités. Une approche plus physique est d'imposer une impulsion aux deux objets lors de la collision. Il est aussi possible de se baser sur une contrainte purement géométrique en altérant la position des objets en collision.

Pénalités

Le principe de la pénalité est d'essayer d'éviter toute collision en imposant une force de répulsion entre deux objets proches. La pénalité la plus simple est d'ajouter un ressort entre les deux objets. On distingue les pénalités linéaires et non-linéaires.

$$f_{lineaire} = k * \left(d_0 - \|\overrightarrow{X_2} - \overrightarrow{X_1}\|\right) * \overrightarrow{N_{1,2}}$$

$$f_{non-lineaire} = \frac{k}{\|\overrightarrow{X_1} - \overrightarrow{X_2}\|}$$

On peut aller plus loin en appliquant des raideurs différentes pour les objets s'approchant et s'éloignant comme le proposent Moore et Wihelms [Moo88].

On peut aussi ajouter un terme de frottement fluide (force dépendant de la vitesse relative des objets) [Tha91]

$$f_{non-lineaire,2} = \left(\frac{k_1}{\|\overrightarrow{X_1} - \overrightarrow{X_2}\|} + k_2 * \left(\overrightarrow{V_1} - \overrightarrow{V_2}\right) .\overrightarrow{N_{1,2}} \right) * \overrightarrow{N_{1,2}}$$

Ces forces n'ont pas de justification physique. Leur choix dépend donc d'autres critères, notamment le réalisme et la stabilité du rendu.

Avantages

- Implémentation rapide
- Calcul rapide
- On modélise facilement divers comportements.

Limites

- Manque de réalisme : les objets peuvent interagir sans se toucher.

Impulsions

Nous travaillons sur des pas d'intégration dont l'ordre de grandeur est la milliseconde. À cette échelle la collision est modélisée physiquement comme une impulsion [Bri03]. On applique ponctuellement une force aux objets en collision. Cette force est calculée selon les lois de la physique (conservation de l'énergie mécanique et de la quantité de mouvement).

Avantages

- Réalisme
- Justification physique.

Limites

- Stabilité du modèle : requiert des pas d'intégration très petits.
- Moins facile à implémenter.

Altération géométrique

Une dernière méthode est d'ajouter une contrainte géométrique au modèle en altérant les positions et les vitesses des objets pour imposer la contrainte[Bar98][Vol95]. Cette méthode peut être couplée aux méthodes précédentes pour gérer les cas où l'impulsion provoque une nouvelle collision, et ceux où la pénalité n'a pas été suffisante pour empêcher la collision.

Une autre approche est proposée ici, il s'agit de repousser, par une modification de la géométrie des triangles, les surfaces en collision. Le modèle se charge alors d'établir l'équilibre entre les forces. La collision étant détectée après coup, deux options sont possibles pour déplacer les éléments vers la position où elle a eu lieu. Soit on utilise les vitesses et les positions courantes pour calculer l'emplacement exact, soit l'on se contente d'imposer une convergence rapide vers une position où les deux surfaces ne sont plus en collision.

Cette deuxième méthode, plus légère, se montre très efficace.

Avantages

- Dans une certaine mesure le réalisme est accru car on évite les abérrations visuelles.

– Le modèle se charge d'établir un équilibre en tenant compte de la contrainte imposée.

Limites

– Stabilité du modèle : requiert des pas d'intégration petits.
– Divergence lorsque les vitesses relatives restent importantes.

4.3.2 Implémentations testées pour la gestion des collisions

Pénalités : force de répulsion

Nous avons testé différentes forces de pénalité. On constate qu'il faut imposer la pénalité avant que la collision n'ait réellement lieu, sous peine de créer des forces entre des surfaces qui n'interagissent pas, mais qui sont proches. Par exemple, si on approche un objet de la face d'un cube à proximité d'une arrête, l'autre face s'appuyant sur l'arrête va se retouver proche de notre objet et interagir avec lui. Ce problème se contourne facilement en donnant une élongation à l'équilibre $d_0 > 0$ au ressort.

Les pénalités linéaires se montrent limitées dès que deux objets sont maintenus à proximité. Si la force qui rapproche les objets est grande, les objets s'interpénètrent, si la force est faible les objets sont maintenus à une distance visible l'un de l'autre.

Une pénalité non linéaire peut s'adapter aux forces extérieures. Elle offre donc un meilleur rendu visuellement, mais le résultat n'est pas parfait si les objets doivent rester en contact.

La méthode de détection des collisions doit être modifiée pour détecter des proximités, en pratique cela revient à augmenter la taille des boîtes englobantes.

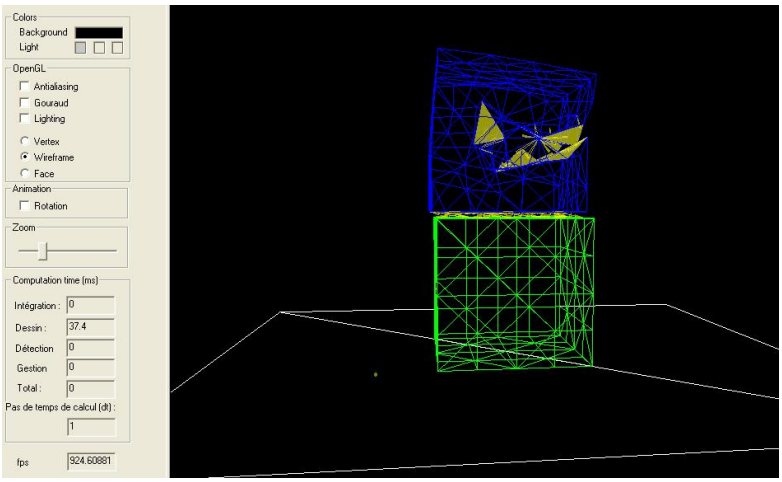

FIG. 4.4 – Méthode des pénalités : visuellement peu satisfaisante en raison des interactions sans contact

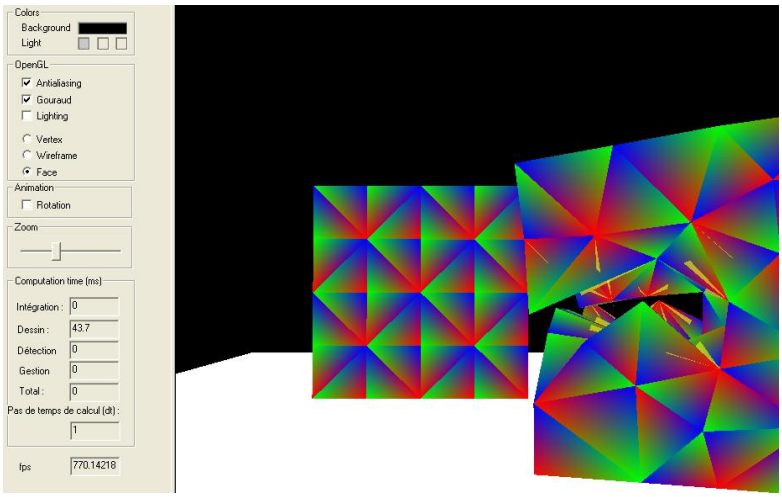

FIG. 4.5 – Methode des pénalités : visuellement peu satisfaisante en raison des interactions sans contact

Altération géométrique

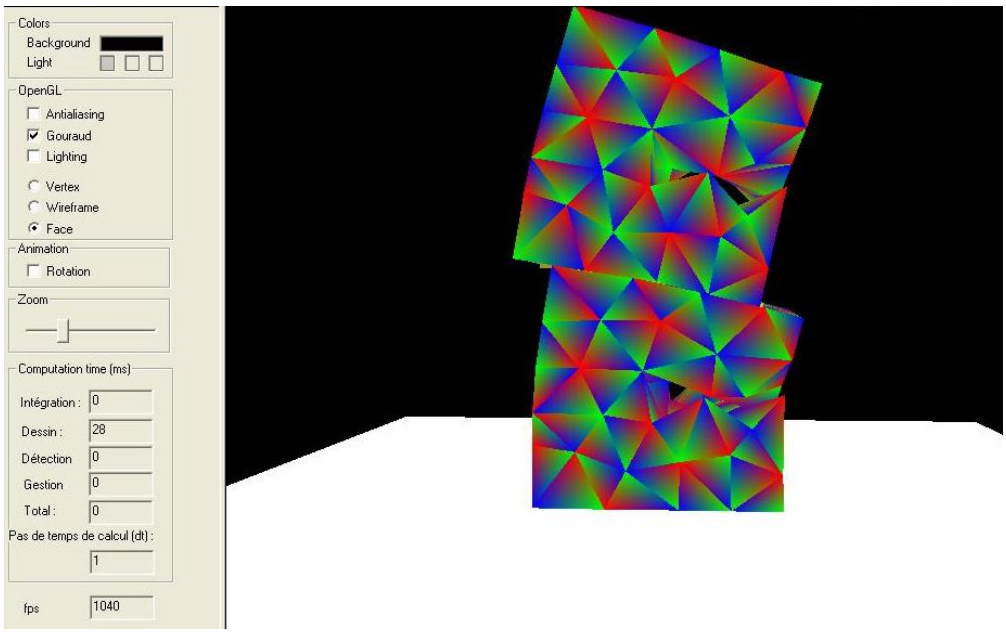

Fig. 4.6 – Auto-collisions gérées pour deux cubes découpés

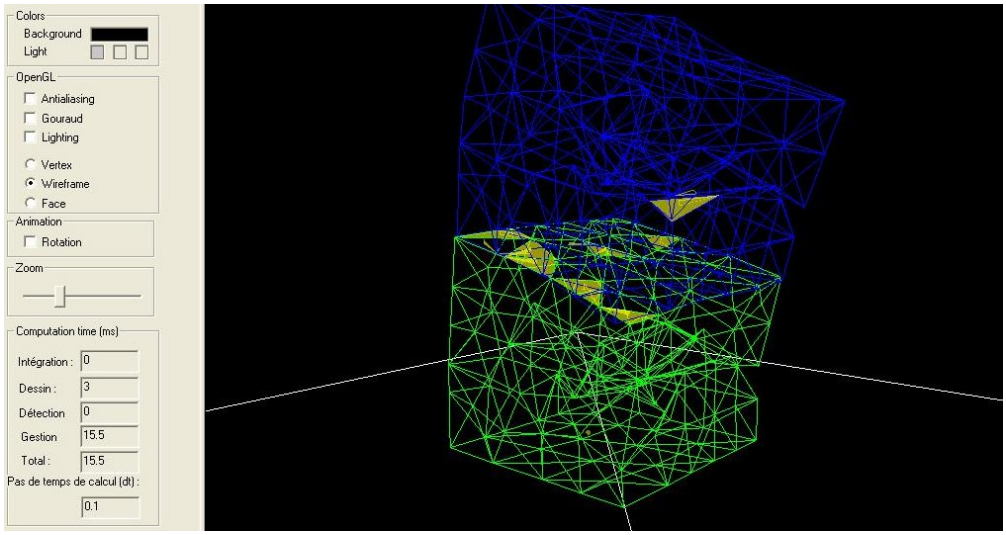

Fig. 4.7 – extra collisions entre les deux

Nous proposons d'utiliser cette méthode comme alternative aux deux méthodes courantes de gestion des collisions.

Altérer la géométrie du modèle lorsqu'une collision a lieu offre un meilleur rendu visuel que la méthode précédente. En effet on obtient de belles surfaces de contact lorsqu'une force rapproche deux objets. On a cependant des problèmes de stabilité lorsque les objets en contact ont des mouvements relatifs importants. Cette méthode ne peut donc s'appliquer que pour des contacts avec frottement.

A chaque étape, on vérifie que la paire de triangles est en collision. Connaissant les normales orientées des triangles, on repousse dans cette direction les différents points de l'autre triangle.

Cette méthode présente deux défauts. Premièrement, des oscillations peuvent apparaître au niveau d'un contact si l'objet est extrait trop vite. Le deuxième défaut apparaît lorsque deux points en lesquels le maillage forme des angles aigus entrent en collision, ces deux points ont artificiellement tendance à converger l'un vers l'autre.

L'aspect visuel est cependant nettement meilleur que celui obtenu avec une méthode par pénalité.

Impulsion

Cette dernière méthode applique une force ponctuellement lorsqu'un contact est detecté. Cette force est calculée à partir des équations de la physique classique (conservation de l'énergie mécanique et de la quantité de mouvement). Si le résultat semble correct visuellement, le modèle diverge assez rapidement à moins d'utiliser des pas d'intégration très faibles. Mais dans ce dernier cas le nombre de boucles à effectuer par seconde devient énorme.

4.3.3 Temps réel et stabilité

La problématique discutée au paragraphe 4.2.2 reste valable ici. Les méthodes présentées sont toutes des méthodes locales et non-récursives, elle sont donc linéaires en fonction du nombre de collisions à traiter.

En conséquence, cette étape n'est pas critique. Toutes les méthodes suivantes sont calculées dans des temps inférieurs à la précision de notre timer (3ms), ce qui limite fortement la portée de cette discussion.

En se basant sur le nombre d'itérations par secondes, on peut les comparer et déduire laquelle des trois est la plus rapide. Cependant il suffit d'augmenter le nombre de primitives du mode pour que la différence devienne négligeable.

Pénalités : force de répulsion

Cette méthode est la moins lourde en temps de calcul. On obtient un nombre d'itérations par seconde supérieur à celui des deux autres méthodes de 20% pour les 596 triangles que l'on peut voir en FIG. 4.6.

Contrainte géométrique

Cette méthode est légèrement moins lourde que la méthode physique en temps de calcul, mais les temps de calcul sont du même ordre de grandeur. On la préfèrera à la précédente car elle offre un rendu plus réaliste.

Impulsion

Cette méthode est la plus gourmande des trois, tout en restant dans des limites raisonnables (au delà des 1000 fps pour un seul cube découpé). Cette méthode impose par contre pour des raisons de stabilité un pas d'intégration plus faible d'un ordre de grandeur (.1 ms avec des vitesse bornées, .01 ms sinon). En conséquence le temps d'une itération doit être divisé d'autant pour réaliser un affichage en temps réel. Il faut donc considérer cette méthode comme beaucoup plus lourde puisqu'elle impose le plus mauvais rapport entre le temps de référence et le temps d'une itération.

Temps réel et stabilité

Le pas d'intégration utilisé est déterminant, car il doit être choisi le plus grand possible d'une part pour des raisons de temps de calcul (si le calcul est plus long que le pas d'intégration, on n'a plus un modèle en temps réel). d'autre part s'il est trop important il va rendre le modèle instable.

4.4 La découpe

Lors de notre présentation des modèles déformables nous avons décrit les critères pour un modèle déformable en particulier ceux qui autorisent les changements de topologie de l'organe pour pouvoir simuler les coupures, déchirures et autres modifications du maillage. Nous allons maintenant nous intéresser à la découpe. Nous verrons les différentes méthodes trouvées dans la littérature pour réaliser la simulation de la découpe du tissu. Il s'agit d'un problème de changement de topologie d'un maillage et les quelques équipes qui ont déjà travaillé sur ce sujet peuvent être regroupées selon trois façons d'aborder le problème. La différence entre ces trois types de méthodes est donc la façon dont le maillage est modifié pour rendre compte de la découpe.

4.4.1 Les trois méthodes pour la découpe

Nous allons exposer les trois méthodes trouvées dans la littérature pour effectuer le changement de topologie.

Retrait des tétraèdres

La première, et la plus simple, des méthodes utilisées consiste à retirer les tétraèdres qui entrent en collision avec l'outil de découpe(Figure 4.8). Une première remarque est de dire que le volume du système n'est pas conservé ou du moins que les quantités perdues s'avèrent importantes vu que le bout du scalpel a une taille bien supérieure à celle du tétraèdre, ce qui entraîne aussi un mauvais rendu graphique. Cependant, elle est simple à mettre en oeuvre car elle ne complexifie pas le maillage (au contraire, elle peut le simplifier) mais il faut que le nombre de tétraèdres soit suffisamment grand et que leur taille soit suffisamment petite.

Cette méthode a été utilisée par Cotin et al. [Cot00]. Forest, lors de sa thèse [For03], a lui aussi utilisé cette méthode car la découpe simulée est celle réalisée par un bistouri à ultrasons qui détruit réellement les cellules. Il propose également un algorithme qui permet de repérer et de corriger les singularités topologiques pouvant apparaître.

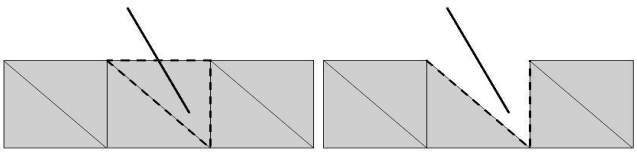

FIG. 4.8 – *Retrait d'un tétraèdre lors de la collision avec l'outil coupant*

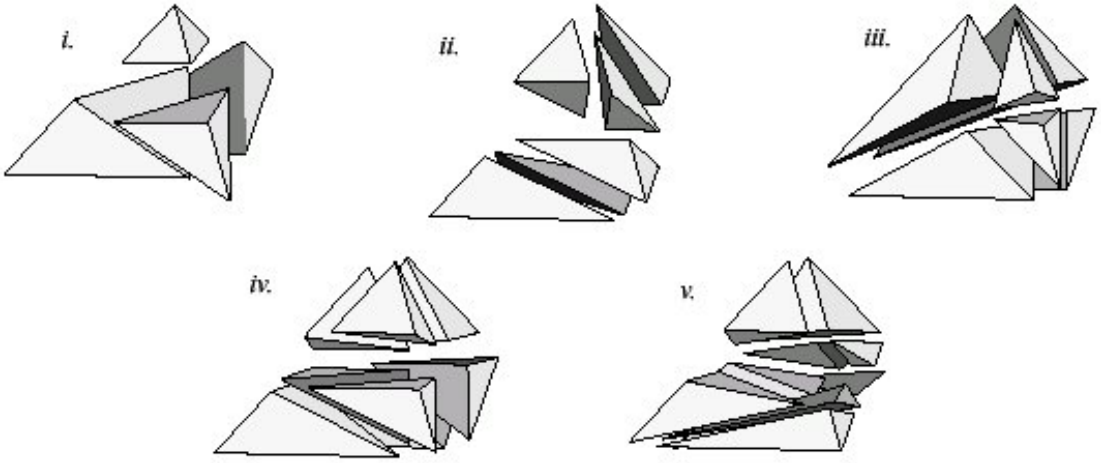

FIG. 4.9 – *Subdivisions minimisant le nombre de tétraèdres créés [Mor01]*

Subdivisions des tétraèdres

Le principe de cette méthode est de subdiviser le tétraèdre qui entre en collision avec le scalpel de manière à ce que le chemin du scalpel sur un tétraèdre corresponde exactement avec un côté commun de deux triangles, faces de deux "sous-tétraèdres" (à la surface de l'organe virtuel, on ne voit que des triangles qui sont une des faces d'un tétraèdre dû au maillage volumique). Les sommets correspondants sont alors dédoublés ce qui engendre l'ouverture entre les deux "sous-tétraèdres"(Figure 4.9). Bielser et al. [Bie00] ont mis en place cette méthode et se sont rendu compte qu'elle était très réaliste graphiquement vu que l'ouverture se fait exactement à l'endroit où l'outil coupant touche le maillage et de plus, il y a conservation du volume du maillage. Cependant cette méthode s'avère être extrêmement coûteuse en temps de calcul car il faut générer les tétraèdres qui décomposent le tétraèdre coupé et le nombre de ces nouveaux tétraèdres augmente au fur et à mesure que l'on découpe. De plus, ces nouveaux tétraèdres peuvent être dégénérés, c'est-à-dire qu'ils ont un volume quasiment nul, ce qui peut entraîner des problèmes lors de la résolution des équations : leur volume proche de zéro peut générer des instabilités numériques.
Mor a également étudié cette méthode lors de sa thèse [Mor01] en imaginant toutes les configurations possibles de l'intersection entre le scalpel et le tétraèdre, ainsi il prévoit à l'avance la décomposition du tétraèdre de manière optimale en termes du nombre de "sous-tétraèdres" créés et de leur taille. Il a également mis en place un algorithme permettant la découpe progressive en créant des subdivisions temporaires qui s'adaptent lorsque l'on continue de couper.

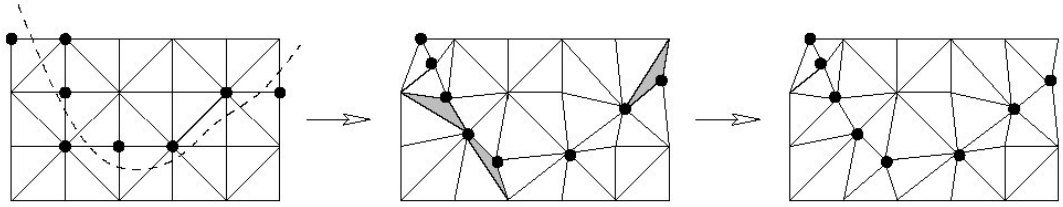

FIG. 4.10 – *Déplacement des sommets pour coïncider avec le chemin du scalpel*

Déplacement des sommets des tétraèdres

Enfin, la dernière méthode possible consiste à raffiner le maillage avant de séparer deux tétraèdres. Nienhuys et van der Stappen [Nie01] [Nie03] ont utilisé cette méthode où les coupures sont réalisées en trois étapes : les faces les plus proches de la trajectoire du scalpel sont d'abord sélectionnées, puis les noeuds de ces faces sont déplacés afin de recouvrir la trajectoire du scalpel, avant d'être séparés pour réaliser la coupure (Figure 4.10). Cette méthode offre l'avantage de ne pas accroître la taille du maillage, l'algorithme conserve donc des performances constantes quel que soit le nombre de coupures effectuées, tout en évitant d'obtenir des contours de coupe irréguliers. Tout comme la méthode précédente, le rendu graphique est très réaliste car il y a aussi coïncidence entre le chemin du scalpel et la coupure, malheureusement dans certaines situations particulières, le processus de projection des noeuds sur la trajectoire du scalpel peut conduire à l'apparition de dégénérescences dans le maillage, et ces cas nécessitent un traitement particulier. Laugier Mendoza et al [Lau03] [Men03] ont également adopté cette façon de voir les choses en développant en plus une méthode pour déterminer les vraies tentatives de découpe par deux critères. Le premier, géométrique, permet d'éliminer les petits déplacements dégénérés et le deuxième, physique, permet d'autoriser la découpe seulement si la force exercée par le scalpel est suffisante.

4.4.2 Les méthodes implémentées

Nous avons choisi de mettre en œuvre deux des trois méthodes déjà citées : la méthode par retrait de tétraèdres et la méthode par déplacement de sommets.

La méthode par retrait de tetraèdres

Comme nous l'avons vu dans l'état de l'art, cette méthode est la plus simple, et elle est adaptée lorsqu'il y a découpe par destruction de matière (bistouri électrique, pince coupante). Elle nécessite également un maillage suffisamment fin pour que le rendu visuel soit correct. Nous allons maintenant nous intéresser à son implémentation qui peut être définie en trois étapes. La première est la détection de collision entre l'outil virtuel et le modèle, ensuite la suppression du tétraèdre qui est entré en collision et enfin les forces

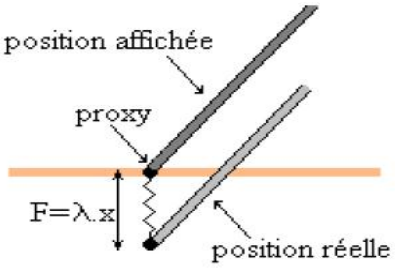

FIG. 4.11 – *La notion de proxy*

ressenties par l'utilisateur (le retour d'effort).
La méthode que nous utilisons pour détecter la collision entre l'outil virtuel et le modèle est une méthode purement géometrique, il s'agit de la collision d'une sphère avec un triangle. La sphère correspond à une sphère englobant le bout de l'outil coupant et le triangle est la face surfacique d'un tétraèdre.
Cette méthode est simple à mettre en œuvre mais va être limitée par le fait qu'il est nécessaire à chaque fois de tester la collision entre la sphère et tous les triangles de surfaces, ce qui peut s'avérer gênant sur un modèle comportant un nombre important de triangles de surface. Nous verrons par la suite des méthodes de détections des collisions plus avancées
Maintenant qu'une collision a été détectée de manière géométrique, nous allons nous intéresser à la déformation causée aux modèles lors de cette collision et au critère qui décidera s'il s'agit d'une tentative de découpe ou non.
La manière de traiter la collision est appellée méthode du God Object, inventée par Zilles et al.[Zil95]. Elle introduit la notion de "proxy" (point de proximité) qui représente la position apparente de l'outil dans la scène par opposition à sa position réelle qui est, elle, déduite directement de la position physique du périphérique. Si l'outil est situé à l'extérieur de l'objet, la position du proxy et celle de l'outil sont identiques. Si la trajectoire de l'outil pénètre l'intérieur de l'objet, le proxy se positionne sur la surface de celui-ci à l'intersection de cette trajectoire (Figure 4.11). Le proxy se déplace alors sur la surface de façon à se rapprocher au maximum de la position réelle de l'outil.
On déforme alors notre objet en déplacant les sommets du triangle de manière à faire se superposer le proxy et le bout de l'outil virtuel. La force envoyée à l'utilisateur est alignée avec la différence entre les positions de l'outil et du proxy et son intensité est proportionnelle à la distance séparant ces deux positions (une fois ramenées dans le référentiel physique du périphérique). Il s'agit donc d'une force ressort de type $F = -k.\Delta x$. En jouant sur le coefficient de raideur k, on peut faire varier la rigidité des objets. On peut même simuler des objets collants en forçant pendant un temps le maintien du proxy sur la surface de l'objet lorsque l'outil en sort. Nous pouvons alors définir un seuil pour la valeur de cette force, seuil qui déterminera si la force exercée sur le tétraèdre est suf-

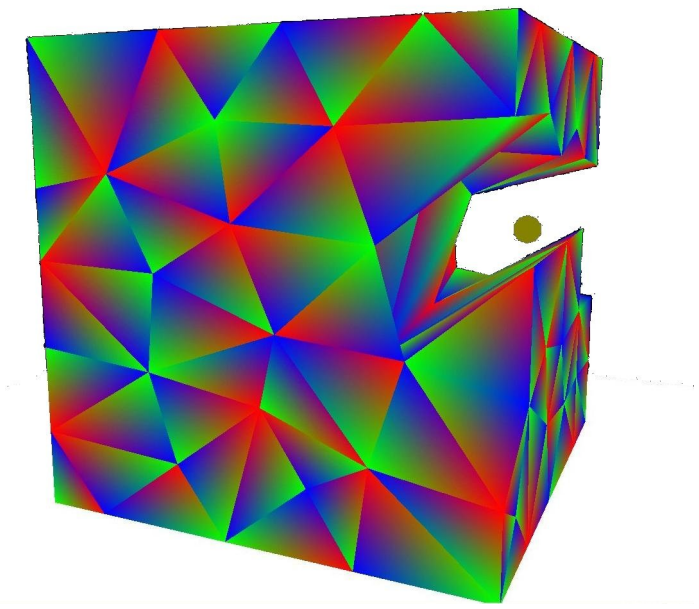

Fig. 4.12 – *La découpe par retrait de tétraèdre*

fisante pour détruire celui-ci.
Dans le cas où la force exercée par le proxy est suffisament forte, le tétraèdre est retiré du modèle.
L'utilisation de cette méthode peut créer des singularités topologiques, définies et étudiées par Forest lors de sa thèse [For03], c'est-à-dire qu'un sommet ou une arête n'appartient plus qu'à deux tétraèdres. Il peut également apparaître des problèmes d'auto-collision, si on laisse tomber dans un champ de gravité le cube découpé (Figure 4.12), les tétraèdres supérieurs de la zone de découpe vont s'écraser sur les inférieurs ce que nous avons présenté avant.

La méthode par déplacement de sommets

Cette méthode, très différente de celle que nous venons de présenter, n'en est pas une concurrente mais une nouvelle manière de voir les choses qui ne s'applique pas dans les même cas. En effet, cette méthode essaye de simuler une ouverture d'un tissu mou, typiquement une découpe effectuée à l'aide d'un scalpel. Dans l'état de l'art, nous avons vu qu'il existait une autre méthode permettant de simuler le même type d'interaction, la méthode par subdivision. Nous avons choisi d'implémenter celle par déplacement de sommet car sa mise en œuvre est plus simple pour un résultat équivalent et même meilleur car le maillage est beaucoup moins dégradé, c'est-à-dire qu'il conserve des tétraèdres plus réguliers et tous de dimension de même ordre de grandeur (la partie suivante sera consacrée à ces problèmes de qualité du maillage). Nous allons commencer par voir comment le maillage est déformé géométriquement de manière à faire correspondre de la meilleure

manière possible le chemin du scalpel et l'ouverture du modèle, puis nous verrons les choix que nous avons réalisés pour le rendu haptique et enfin nous soulèverons le problème de l'apparition des tétraèdres dégénérés.

L'aspect géométrique Dans ce paragraphe, nous allons voir comment sont déplacés les sommets de certains tétraèdres de manière à ce que le chemin du scalpel puisse être représenté par une succession d'arêtes des triangles de surface (Figure 4.13). Puis une fois qu'ils ont été déplacés, on dédouble les sommets ce qui fait apparaître l'ouverture et enfin la dernière chose à effectuer c'est d'aligner les sommets en profondeur.

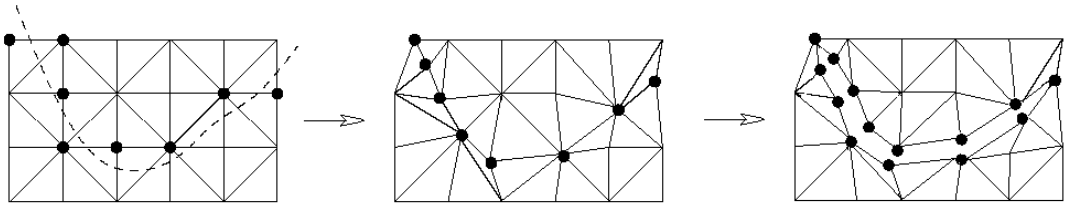

FIG. 4.13 – *algorithme géomètrique de l'ouverture d'un maillage*

Initialisation Le critère géomètrique conduisant au début de la découpe est la sélection de trois sommets (Figure 4.14). Cependant, il est nécessaire d'avoir un sommet qui n'appartienne pas au même triangle que les deux autres, et si cette condition est réalisée, l'ouverture est possible et la phase de découpe peut commencer.
Lors du premier contact de l'outil avec les triangles de surface, le sommet le plus proche

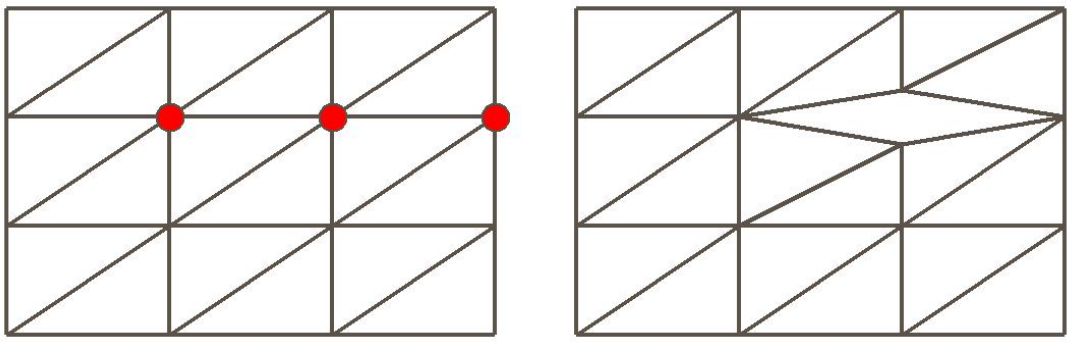

FIG. 4.14 – *ouverture minimale d'un maillage*

du point d'intersection est repositionné à l'endroit de l'intersection, puis commence alors

l'algorithme de découpe qui sera précenté ci-dessous, et dès que la condition sur les trois premiers sommets présentée précédemment est vérifiée, l'ouverture est réalisée, c'est-à-dire que le sommet du milieu est dédoublé. Ensuite à chaque fois qu'un nouveau point est sélectionné par l'algorithme, il est a priori considéré comme étant la fin de la découpe et donc le point précédent est dédoublé.

Algorithme de projection L'algorithme de projection est la phase durant laquelle des sommets sont déplacés de manière à ce que le chemin du scalpel corresponde au mieux avec des arêtes de triangles de surface(Figure 4.15).
Considérons une des étapes de la figure précédente, au moment où un point a été dé-

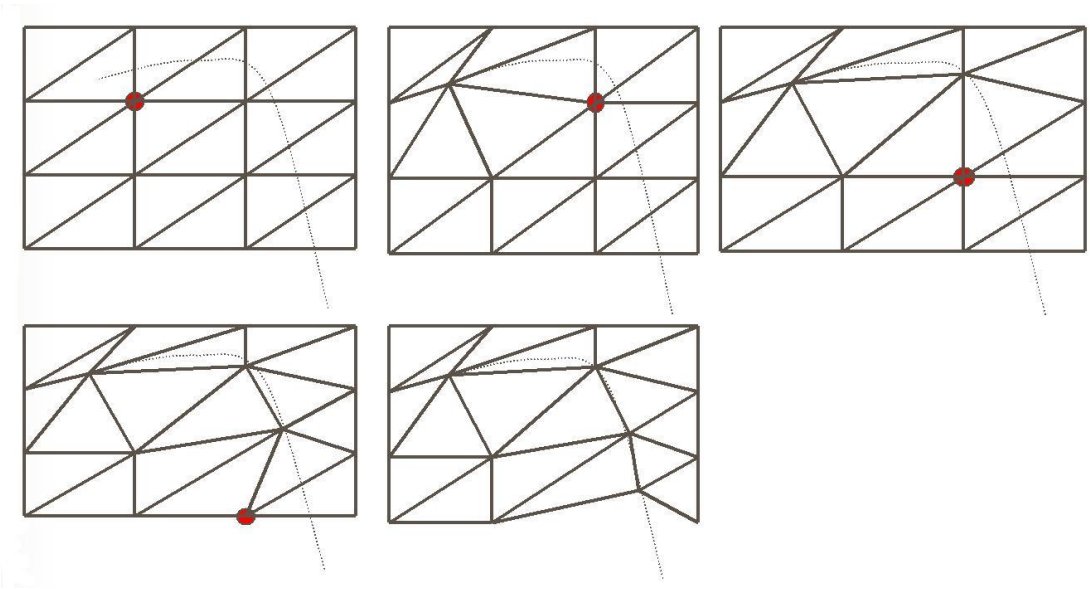

FIG. 4.15 – *Projection des sommets le long du chemin du scalpel. A chaque étape, un nouveau sommet est selectionné, le point rouge et il est replacé sur le parcours.*

placé. L'outil virtuel continue son chemin, tant que le point précédemment déplacé reste le point le plus proche de l'intersection du scalpel et de la surface, rien ne se passe. Il arrive un moment où le scalpel est plus proche d'un nouveau point, on vérifie que ce point est bien dans la continuité de la trajectoire du scalpel et qu'il n'appartient pas aux mêmes triangles que les deux derniers sommets déplacés, si ces conditions sont réalisées, le point est stocké et sera le prochain point déplacé. Il ne sera effectivement déplacé qu'au moment où l'outil ne s'approchera plus de ce point mais s'en éloignera.
Un fois que l'on a déplacé le point, juste avant de dédoubler le point qui avait été déplacé auparavant, il reste une opération à effectuer, c'est de déplacer les sommets en profondeur pour que l'ouverture soit bien dans la direction du chemin du scalpel.

Traitement de la profondeur Avant de dédoubler un sommet, ses sommets voisins à l'intérieur de l'objet doivent être repositionnés pour que le rendu visuel de l'ouverture soit cohérent avec le chemin du scalpel. Les sommets en profondeur, c'est-à-dire les quatrièmes sommets des tétraèdres dont une arête représente un bord de l'ouverture, sont projetés sur le plan défini par le parcours de l'outil. Si cette opération n'est pas réalisée, il est possible que les sommets en profondeur soient assez loin du plan du chemin du scalpel et on obtient alors un mauvais rendu graphique (Figure 4.16).
Cette étape s'avère particulièrement importante lorsque le maillage est irrégulier et gros-

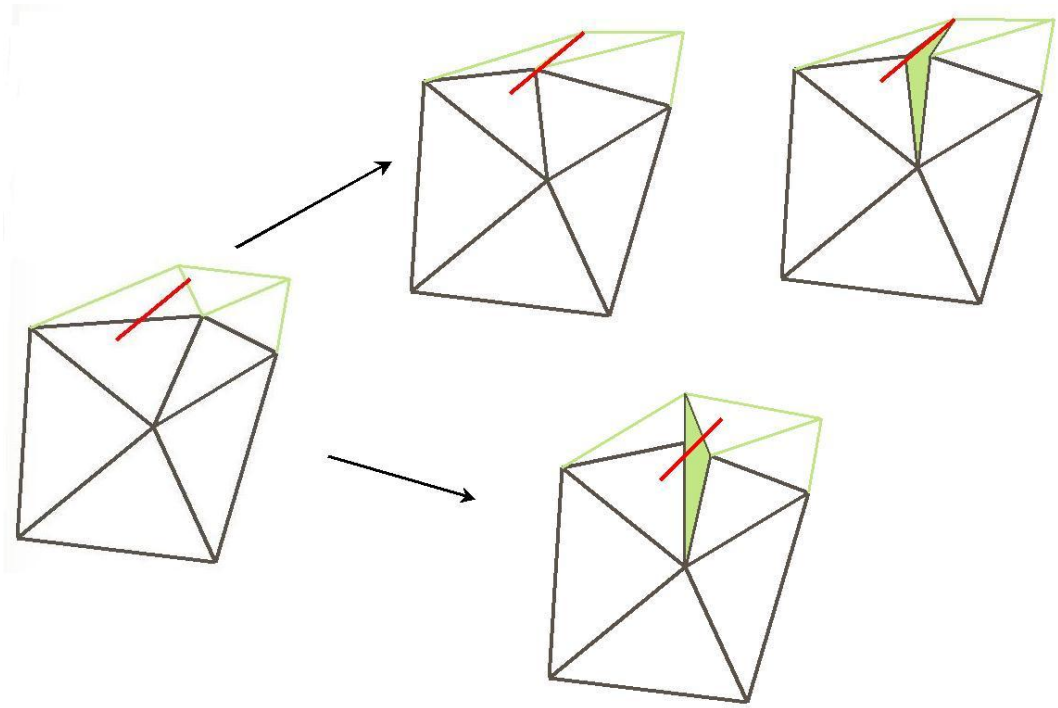

FIG. 4.16 – *Dans le schéma du haut on voit comment les sommets en profondeurs sont projetés sur le plan défini par le scalpel ; en bas on voit un cas critique où le rendu est mauvais si la projection n'est pas réalisée.*

sier. Dans le cas contraire, les déplacements à effectuer sont très faibles par rapport à l'ordre de grandeur des longueurs des arêtes et ils doivent même être surveillés car ils sont sources d'instabilités numériques.

Les dégénérescences

Nous venons de présenter deux algorithmes de découpe, mais il reste encore un problème important à traiter pour la méthode de découpe par déplacement de sommets. Il s'agit des dégénérescences qui apparaissent lorsque nous déformons les tétraèdres de manière à ce qu'une de leurs arêtes corresponde avec le chemin de la découpe. Un tétraèdre

est dit dégénéré s'il possède un volume quasi nul alors que l'une au moins de ses arêtes n'est pas de longueur nulle. L'apparition de ces tétraèdres dégrade fortement la qualité du maillage. Les tétraèdres dégénérés nuisent au comportement du modèle (Figure 4.17) et peuvent également apporter des instabilités numériques. Nous allons commencer par présenter les critères permettant de juger de la qualité d'un maillage et des tétraèdres, ensuite nous examinerons les différentes manières qui permettent d'éliminer ces dégénérescences et nous comparerons la qualité du modèle obtenue par les deux approches.

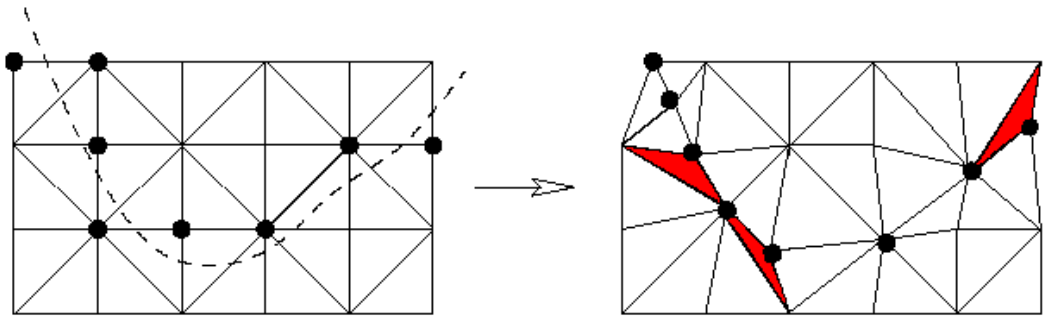

FIG. 4.17 – *Apparition de dégénérescences lors de la phase de déplacement des sommets.*

La qualité d'un maillage Pour un maillage tétraèdrique, il y a deux niveaux auxquels on juge la qualité d'un maillage. Au niveau des tétraèdres, on juge la qualité de celui-ci en le comparant au tétraèdre équilatéral. On trouve ainsi dans la littérature beaucoup de critères géomètriques permettant cette comparaison. A un niveau plus global, en considérant l'ensemble des tétraèdres, un bon maillage sera un maillage très homogène. Il ne suffit pas que les tétraèdres soient équilatéraux, il est également nécessaire que leur taille soit voisine. Nous allons commencer par présenter les différents types de tétraèdres dégénérés pouvant apparaître lors de notre algorithme de découpe.

Les différents types de tétraèdres singuliers Forest [For03] a introduit une classification des tétraèdres singuliers, ils peuvent être regroupés en deux types (Figure 4.18).

Type I : l'une (au moins) des arêtes est très courte par rapport aux autres. On peut encore distinguer entre les tétraèdres ayant exactement une, deux et trois petite(s) arête(s).

Type II : les quatre sommets, sans être proches les uns des autres, sont presque coplanaires.

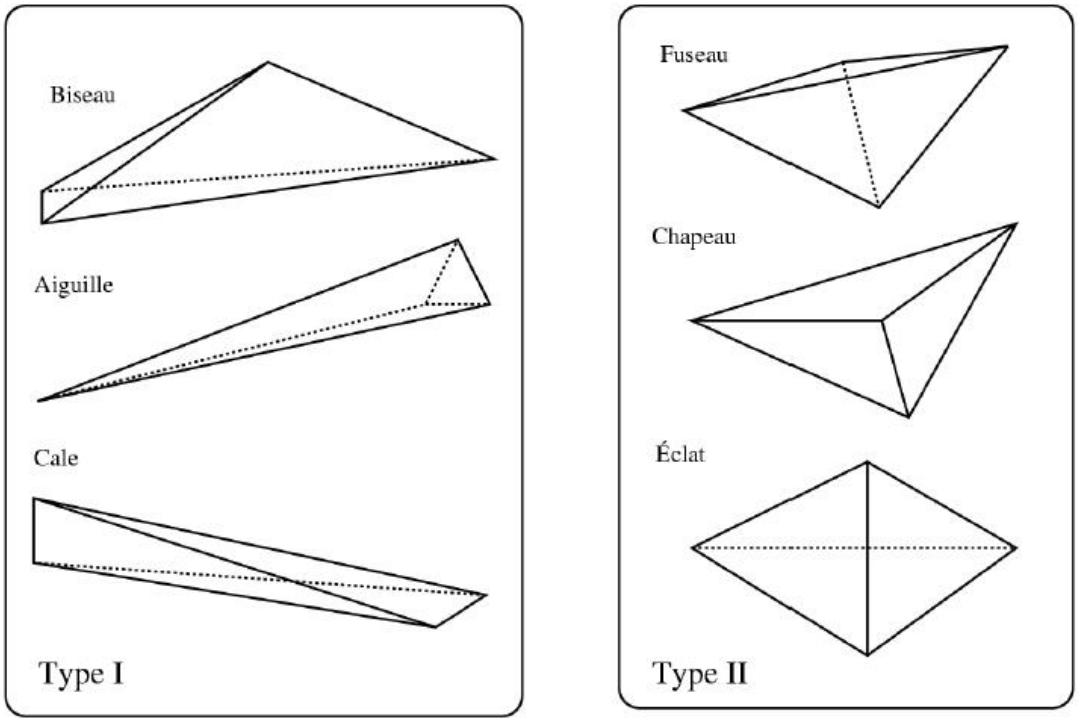

FIG. 4.18 – *Différents tétraèdres singulier de type I ou II [For03]*

Les critères de qualité d'un tétraèdre Il s'agit de trouver des critères permettant de repèrer ces singularités, il en existe un grand nombre car il n'existe pas de mesure objective de la qualité d'un tétraèdre. Tous ces critères utilisent des mesures géomètriques du tétraèdre :

- l_i, la longueur de l'arête i.

- s_i, la surface de la face i, elle vaut $\sqrt{3}$ pour l'équilatéral unitaire.

- v, son volume, il vaut $\frac{\sqrt{2}}{12}$ pour l'équilatéral unitaire.

- ρ_{int}, le rayon de la sphère inscrite, $\rho_{int} = \frac{3.v}{\sum_i s_i}$, il vaut $\frac{\sqrt{6}}{12}$ pour l'équilatéral unitaire.

- ρ_{int}, le rayon de la sphère circonscrite, $\rho_{int} = \frac{\sqrt{(a+b+c)(-a+b+c)(a-b+c)(a+b-c)}}{24.v}$ ou a,b et c sont les produits des longueurs des arêtes opposées, il vaut $\frac{\sqrt{6}}{4}$ pour l'équilatéral unitaire.

- θ_i, l'angle solide d'un sommet, il correspond à la surface formée par la projection

de ses trois sommets adjacents sur la sphère unité.

- ϕ_i, l'angle dièdre entre deux faces du tétraèdre, il est l'angle entre l'intersection de ces deux faces par un plan perpendiculaire à l'arête commune.

On combine alors ces mesures pour obtenir des rapports compris entre 0 et 1, 1 pour le tétraèdre équilatéral, ou alors on mesure juste le minimum ou le maximum de ces valeurs et ceci nous permet d'obtenir des critères mesurant la qualité du tétraèdre. On peut en imaginer une infinité mais certains seront plus ou moins adaptés à un ou plusieurs types de tétraèdres singuliers. Voici quelques exemples de ces critères :

- $\frac{\min_i l_i}{\max_i l_i}$, le rapport r entre la longueur de l'arête la plus longue et celle de l'arête la plus courte, il ne discrimine pas les tétraèdres singuliers de type éclat.

- $\frac{12}{\sqrt{6}}.\frac{\max_i l_i}{\rho_{int}}$, le facteur multiplicatif sert juste à ce que toutes les valeurs du rapport soit comprises entre 0 et 1 avec 1 pour l'équilatéral unitaire.

- $3.\frac{\rho_{int}}{\rho_{ext}}$, Le rapport normalisé entre le rayon des sphères circonscrite et inscrite.

- $\alpha.\min_i \theta_i$, avec $\alpha = 6.\arcsin(\sqrt{3}/3) - \pi$ l'angle solide des sommets du tétraèdre régulier. Le calcul des θ_i se révèle coûteux car il passe par des fonctions trigonométriques inverses. On minimise plutôt le $\sin\theta_i/2$ car il se calcule directement à partir des longueurs (voir [For03] pour plus de détails).

- $\alpha.\min_i \phi_i$, où $\alpha = \pi - \arccos(-1/3)$ est un coefficient de normalisation correspondant à l'angle dièdre entre deux facettes du tétraèdre isocèle. Il ne discrimine pas les tétraèdres singuliers de type aiguille.

Péquignot[Peq99] a étudié ces différents critères et il s'avère qu'ils sont souvent équivalents à part pour certains qui distinguent plus particulièrement un type de singularité. Nous avons pour notre part choisi celui du rapport des sphères inscrite et circonscrite **RRS** car il s'avère détecter tous les types de dégénérescences, et également celui des rapports des longueurs minimale et maximale des arêtes car il sera utile lors que l'on utilisera la méthode d'écrasement décrite ci-dessous.

Les différentes approches d'élimination des tétraèdres dégénérés

Approche par remaillage Dans la littérature, nous pouvons trouver trois façons distinctes de résoudre ce problème, c'est-à-dire la suppression de ces tétraèdres. La première fut mise en oeuvre par Nienhuys [Nie03], elle consiste à effectuer un remaillage de Delaunay dans la zone autour du tétraèdre. Un maillage tétraèdrique est dit de Delaunay lorsque la sphère circonscrite de chaque tétraèdre du maillage ne contient aucun

sommet autre que ceux de ce même tétraèdre (Figure 4.19). Les maillages de Delaunay possèdent un grand nombre de qualités, entre autre leur unicité étant donné un ensemble de sommets donnés (sauf cas particuliers, lorsque plusieurs sommets sont cocycliques) et leur facilité de construction. De plus, si les sommets sont bien répartis, la qualité des tétraèdres générés est relativement acceptable, même s'il n'y a pas de garantie théorique à cela. En particulier, les tétraèdres dégénérés de type éclat sont tout à fait admissibles par les triangulations de Delaunay, ce qui fait que les mailleurs traditionnels sont rarement Delaunay, ou au moins disposent d'autres critères permettant l'amélioration du maillage.Cette méthode pose cependant certains problèmes : le remaillage que l'on souhaite effectuer doit en effet préserver la surface du maillage ainsi que les éventuelles sous-structures qui le composent, on parle alors de triangulation de Delaunay contrainte, et cette exigence peut parfois empêcher la convergence des algorithmes de remaillage, surtout dans le cas de contraintes présentant des angles très faibles. C. Paloc [Pal02] propose cependant une méthode de raffinement dynamique de maillage volumique utilisant les propriétés de Delaunay. Nous avons pour notre part développé notre propre version de l'algorithme de remaillage adapté à notre modèle et aux contraintes citées plus haut.

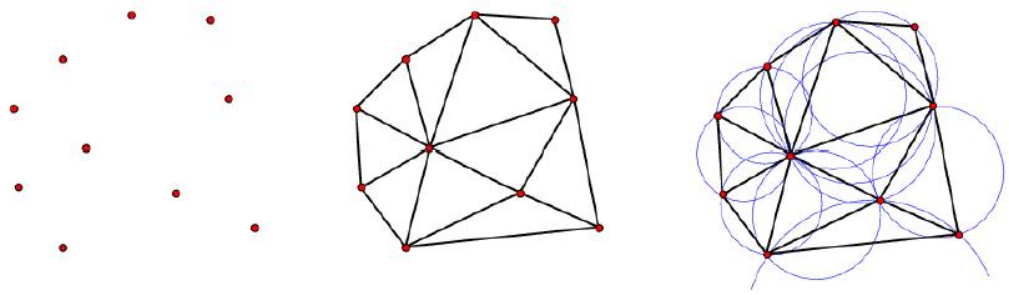

Fig. 4.19 – *A gauche, un nuage de points. Au centre, sa triangulation de Delaunay. A droite, on a représenté l'ensemble des cercles circonscrits.*

Approche par écrasement La deuxième approche développée par Mendoza [Men03] est beaucoup plus simple : il considère que les tétraèdres dégénérés par la méthode de déplacement de sommets sont tous de types I et II, et donc qu'il peut trouver une arête du tétraèdre (toujours en surface) dont la taille soit bien plus petite que celle des autres et il "aplatit" alors le tétraèdre en annulant la longueur de cette arrête (Figure 4.20). Cette méthode nécessite parfois de supprimer plusieurs tétraèdres, en effet si l'arête dont la longueur doit être annulée est en commun avec d'autres tétraèdres, ils devront être supprimés. De plus, le fait d'aplatir le tétraèdre engendre une augmentation de la taille des tétraèdres voisins. Nous avons également implémenté cette méthode, et dans le prochain paragraphe nous allons présenter les résultats obtenus, c'est-à-dire l'évolution des critères de qualité présentés précédemment après une phase de découpe qui aura fait

apparaître des tétraèdres dégénérés et qui seront supprimés par cette méthode ou par celle du remaillage.

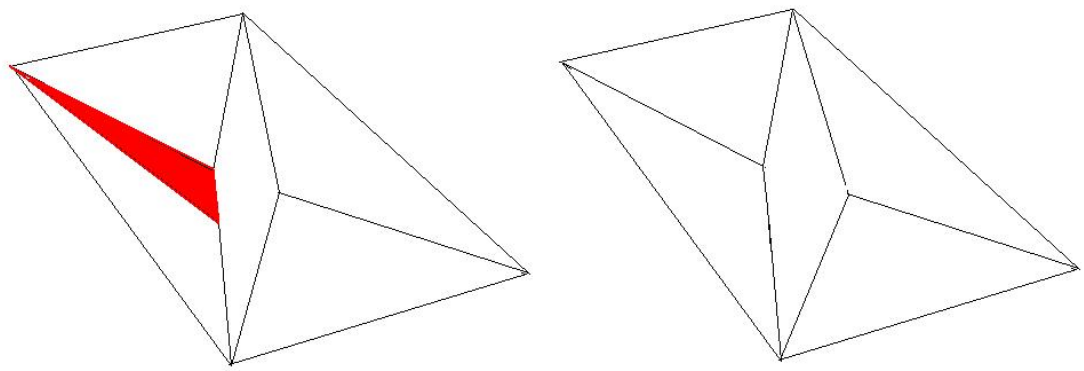

FIG. 4.20 – *On a représenté la face du tétraèdre apparaissant en surface de l'objet et qui contient la longueur qui est annulée à gauche. A droite, on présente le résultat*

Qualité des méthodes utilisées Nous allons donc maintenant présenter la qualité des maillages obtenue après les deux méthodes de suppression des tétraèdres singuliers, l'aplatissement et le remaillage. Nous voyons que la qualité du maillage obtenu par les deux méthodes n'est pas la même. Nous nous sommes alors intéressés à la façon dont les tétraèdres se déformaient par rapport aux tétraèdres initiaux par les deux méthodes ; pour illustrer cette différence de dispersion des valeurs des critères de qualité, nous avons représenté les valeurs des deux critères de qualité en fonction du volume pour chaque tétraèdre (Figure 4.21).
Nous verifions bien que la méthode par remaillage disperse beaucoup moins que l'autre, c'est-à-dire que la qualité des tétraèdres et du maillage est meilleure lorsque l'on remaille la zone aux alentours du tétraèdre singulier. Nous la considérons donc comme la meilleure des deux méthodes étudiées.

4.5 Conclusion

Nous nous sommes intéressés à la détection et la gestion des collisions et des auto-collisions. Notre modèle prend maintenant en compte les collisions et auto-collisions de façon efficace. Sweep and prune, l'algorithme de détection soigneusement choisi parmi les algorithmes disponibles, permet une détection efficace. Son choix a été motivé par des considérations de complexité algorithmique. Il restera donc efficace si on étend la précision du modèle sur une machine plus puissante.

Après avoir détecté une collision, il faut la gérer. Les deux algorithmes classiques de gestion des collisions ont éte testés. La méthode des pénalités s'est montrée trop peu réaliste et a donc été abandonnée. La méthode des impulsions impose des pas de temps trop faibles, ce qui nous éloigne de notre problématique de temps réel, imposée par l'objectif de réaliser un modèle pour faire de la simulation chirurgicale. Un troisième algorithme est donc proposé. En imposant des contraintes géométriques, il permet d'avoir un rendu à la fois rapide et visuellement convaincant. Sa principale limite est de ne marcher qu'avec des frottements importants, ce qui ne nous gêne que peu car les tissus vivants que nous souhaitons modéliser entrent dans cette catégorie.

Nous avons implémenté deux méthodes simulant deux types de découpe. La première correspond à une découpe effectuée par un bistouri électrique ou par tout autre instrument détruisant effectivement la matière, la deuxième, adaptée lorsque la découpe est réalisée par un outil tranchant, effectue une ouverture dans le modèle numérique. Dans la seconde approche nous nous sommes aperçus que le maillage devait être raffiné après la phase de découpe pour obtenir un rendu visuel et comportemental réalistes. Nous avons donc implémenté deux méthodes effectuant cette opération et identifié celle qui s'avère plus performante, c'est-à-dire moins dégradante pour le maillage. Il s'agit de la méthode où l'on effectue un remaillage dans la zone de la découpe tout en traitant les dégénérescences créées.

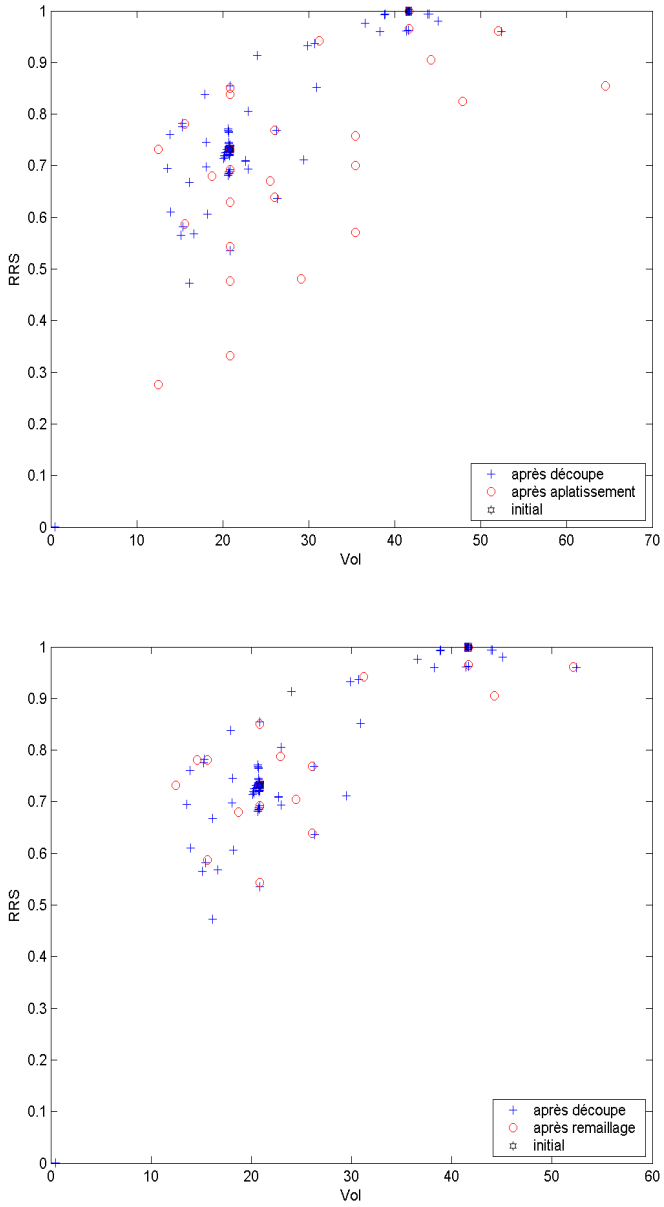

FIG. 4.21 – *Répartition des valeurs du critère de qualité RRS en fonction du volume, pour les deux méthodes de traitement des dégénérescences étudiées, par aplatissement (en haut) et par remaillage (en bas). On voit la supériorité de cette deuxième méthode.*

Chapitre 5

Système haptique dédié à la chirurgie virtuelle

Dans ce chapitre, nous présentons nos travaux de conception et de réalisation du système de retour d'effort dédié à la chirurgie, le « trocart actif ». Il s'agit de l'instrumentation du trocart, outil indispensable pour la chirurgie mini-invasive afin d'assurer au maximum que les outils de chirurgie ne heurtent pas la peau à l'endroit de l'incision. Nous l'appelons « trocart actif ». Il s'agit d'un système de retour d'effort à 5 degrés de liberté.

5.1 Les interfaces haptiques

En évoquant le terme « haptique »qui vient du grec Haptesthai et qui signifie sentir par le toucher, on se réfère souvent à l'utilisation des mains pour des sensations tactiles qui stimulent des excitations de la peau et englobent des modalités très variées comme la température, la texture ou le contact. Viennent s'ajouter à ces sensations celles des efforts qui désignent l'exercice de forces sur les membres du corps et qui contraignent le mouvement.

La connaissance du sens haptique humain est une étape fondamentale dans l'élaboration d'un environnement virtuel à retour haptique. Elle permet en premier lieu de mieux concevoir les périphériques haptiques. Les interfaces haptiques actuelles permettent à l'utilisateur de toucher, sentir et manipuler des environnements virtuels, ce qui procure une sensation d'interaction multimodale considérable. Faire la synthèse haptique d'objets virtuels (rigidité, texture, et forme), requiert un compromis entre la complexité du calcul requis pour obtenir les paramètres haptiques issus de l'interaction et des aspects temps réel du rendu haptique. Ce compromis prend en compte les aspects psychophysiques de la perception haptique ainsi que le type et la qualité de l'interface haptique utilisée.

Un système haptique est un système mécanique qui permet de simuler de façon virtuelle le contact entre un outil et un objet.

L'interaction haptique peut être définie comme la combinaison étroite de deux fonctionnalités, une modalité sensorielle de nature tactilo-kinesthésique qui met en jeu une perception tactile couplée à des mouvements exploratoires de l'organe de perception, et une capacité « d'action physique virtuelle »dans l'espace d'interaction.

Un système haptique a donc pour fonction d'une part de permettre de guider un outil virtuel dans le monde simulé, et d'autre part de transmettre en permanence des forces aux mains ou aux doigts afin de donner l'illusion de « toucher »quelque chose.

Historiquement, les interfaces haptiques sont issues des techniques de télé-opération. Les bras maître /esclave utilisés pour la télé-opération, ont connu une évolution considérable à travers les années. Les plus anciens sont les bras utilisant des transmissions mécaniques à câble pour assurer le couplage des bras maîtres.

Les interfaces haptiques sont aussi utilisées pour interagir physiquement avec un environnement virtuel. Les travaux menés à l'université de Caroline du Nord à Chapel Hill [Tay99], montrent que les applications de réalité virtuelle utilisant le retour d'effort remontent au début des années 70. Après avoir utilisé des organes à 2 Degrés de Liberté (DDL), les chercheurs ont par la suite utilisé des bras maître à 6 DDL à retour d'effort pour des applications pharmaceutiques (i.e. pour simuler l'interaction de molécules).

Contrairement aux bras maîtres développés dans le cadre de la télé-opération, les interfaces haptiques actuelles privilégient pour la plupart le confort au détriment d'un espace de travail important et la sensibilité au détriment d'une forte capacité en effort. On notera d'autre part qu'elles ont bénéficié de l'expérience acquise sur les bras maîtres. On trouve par conséquent sur le marché des interfaces assez variées, entre celles qui utilisent une structure parallèle, structure à câble ou à lévitation magnétique. Le choix reste guidé essentiellement par l'application.

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 entendue orientée vers les manettes de jeux vidéo (Microsoft, Immersion). On peut classer les différents systèmes suivant plusieurs manières. Les systèmes haptiques considérés peuvent être génériques ou bien orientés vers une application bien définie.

Par exemple un « gant haptique »permet de simuler les efforts ressentis sur les doigts et la main lorsqu'on saisit un objet [DLB97] [PBB99]. Il peut être utilisé pour la rééducation de la main et du poignet [BDLLG97]. D'autres dispositifs ont un champ plus important dans la mesure où ils permettent simplement de ressentir une force en un point de l'espace.

Créer artificiellement une sensation tactile réaliste nécessite le calcul des forces et des vibrations impliquées à une fréquence suffisamment élevée. Typiquement, la perception humaine met en jeu des fréquences allant de 300Hz pour des corps mous, à 10KHz pour des objets rigides [Bro91] [Ell96]. De telles fréquences peuvent facilement être atteintes lorsque l'on manipule des objets virtuels simples sans dynamique interne ; cela devient rapidement hors de portée de la technologie informatique grand public actuelle, lorsque l'on considère des scènes virtuelles mettant en jeu des objets complexes et des phénomènes dynamiques variés comme ceux engendrés par des corps déformables (comme c'est le cas pour la simulation d'organes humains par exemple). Le principal obstacle réside alors dans le processus de simulation dynamique qui fonctionne généralement à une fréquence compatible avec le sens de la vision (soit de l'ordre de 30Hz), du fait de la complexité des modèles des algorithmes mis en jeu. Cette difficulté est contournée au niveau de l'interaction haptique au moyen d'un modèle informatique approprié.

5.2 Les interfaces haptiques pour l'apprentissage du geste chirurgical

Des travaux de recherches ont déjà abouti à une commercialisation de plusieurs systèmes haptiques. Parmi ces interfaces on peut distinguer les dispositifs haptiques universels souvent utilisés pour le développement des applications génériques de réalité virtuelle ou pour validation des applications tel le Phantom Desktop de la société SensAble Technologies ou le Virtuose 6D de Haption (figure 5.1), ces interfaces sont munies d'un préhenseur de type stylo ou poignée comme l'Omega de la société ForceDimension (figure 5.2). On distingue aussi interfaces haptiques dédiées comme des joysticks à retour d'effort pour certains jeux vidéo et les interfaces haptiques pour les applications médicales. Une autre classification des systèmes haptiques est selon les différentes architectures possibles (Joysticks, bras articulés, pantographes, etc.). L'ouvrage [Fus06] présente d'amples informations sur les systèmes à retour d'effort, leurs principes, classifications ainsi que des nombreux prototypes qui ont été développés au cours des dernières années. Nous nous intéressons dans cette section aux interfaces haptiques dédiées à la chirurgie endoscopique.

L'utilisation d'interfaces haptiques dédiées à la chirurgie endoscopique revient au fait que les mouvements de l'opérateur sont contraints par la présence d'un trocart au niveau de la peau du patient.

Fig. 5.1 – Le Phantom Desktop de SensAble Technologies et le Virtuose 6D de Haption

Fig. 5.2 – L'Omega.3 de la société ForceDimension

Fig. 5.3 – A gauche : le Virtual Laparoscopic interface. Au centre : le Laparoscopic Impulse Engine. A droite : Le Laparoscopic Surgical Workstation

Le Laparoscopic Impulse Engine LIE (figure 5.3 au centre) est un simulateur de laparoscopie à retour d'effort. C'est une interface très utilisée dans les applications médicales. Il contient 5 articulations toutes de natures différentes. Il permet la restitution d'effort selon 5 DDL, une bande passante de 650 Hz (axe de translation) et 120 Hz (axes de rotation) ainsi qu'un espace de travail, une résolution et effort maximum spécifiques pour chaque articulation.

Avant le LIE, Immersion a commercialisé le Virtual Laparoscopic interface VLI (figure 5.3), c'est aussi une interface de simulateur de laparoscopie sauf qu'elle n'est pas motorisée c'est-à-dire à DDL passif. A l'instar du LIE, le VLI contient 5 articulations toutes de types différents. Chacune de ces articulations a ses propres caractéristiques. Le LIE est assez volumineux, il occupe une grande place rendant difficile la création de l'environnement réel d'une opération chirurgicale qui exige dans la plupart des cas l'utilisation de plusieurs outils de chirurgie. Immersion a développé le Laparoscopic Surgical Workstation regroupant deux outils similaires à LIE dans le même bâti (figure 5.3 à droite).

Le HIT Force Feedback Device est une interface qui a été développée pour KISMET, il a été ensuite commercialisé par la société Select-it sous le nom VEST System One « VSOne »(figure 5.4).

Les systèmes haptiques existants sont en majorité volumineux et occupent un grand espace ce qui rend délicat la création de l'environnement d'une vraie opération chirurgicale, le VSOne a la possibilité d'utiliser plusieurs outils de chirurgie en même temps (au maximum 3 outils + une caméra) mais sous une configuration fixe de l'espace de travail. Il en est de même pour le LapMentor de la société Simbionix (figure 5.4). Notons que ces deux systèmes sont inclus dans un simulateur de chirurgie et il ne s'agit pas d'un système haptique ouvert pour toute application.

Un autre système réussit à simuler ou à représenter le même environnement d'une opération chirurgicale spécifique, Papadopoulos et al. présente dans [Pap02] un système à retour d'effort pour les opérations d'urologie. L'extension de ce système vers d'autre type d'opération exige une modification de l'interface du système, voire même une mo-

FIG. 5.4 – A gauche : Le HIT Force Feedback Device. Au centre : le simulateur Vest Systeme One. A droite : Le simulateur LapMentor

dification de la conception mécanique.

5.3 Conception d'une nouvelle interface : trocart actif

Dans les opérations endoscopiques le chirurgien pratique des micro incisions par lesquelles il fait passer des tiges cylindriques appelé trocarts (figure 5.5). Ces trocarts, comme les montre la figure 5.6, en nombre variable de 1 à 5, sont mis en place et servent au passage de fins instruments ainsi que l'endoscope qui permettent au chirurgien et à ses aides d'assurer l'opération en contrôlant les gestes chirurgicaux sur le moniteur.

L'utilisation des trocarts est indispensable pour la chirurgie mini-invasive afin d'assurer au maximum que les outils de chirurgie ne heurtent pas la peau à l'endroit de l'incision.

L'instrumentation du trocart pour restituer l'effort, et non pas l'outil chirurgical en soi, permet une richesse de choix de l'outil ainsi que plus de réalisme pour l'apprentissage, le chirurgien a la possibilité des changer l'outil en pleine opération. Une fois inséré, le nouvel outil sera détecté et identifié. Nous avons optimisé la conception de ce trocart que nous appelons **"trocart actif"** afin de permettre l'utilisation de plusieurs outils de chirurgie dans la même opération de chirurgie virtuelle et évidemment afin d'assurer le bon fonctionnement d'un système haptique pour garantir un bon rendement tactile.

5.3.1 Les gestes du chirurgien - analyse du besoin

Avant de parler du retour d'effort, notre système doit assurer la fonction d'une interface de chirurgie qui nous permet de placer le chirurgien dans un environnement virtuel très proche de l'environnement d'une vraie opération chirurgicale. Ce système doit donc permettre à un utilisateur de réaliser les gestes que les chirurgiens pratiquent lors des opérations chirurgicales.

Fig. 5.5 – Les trocarts et Les outis de chirurgie endoscopique

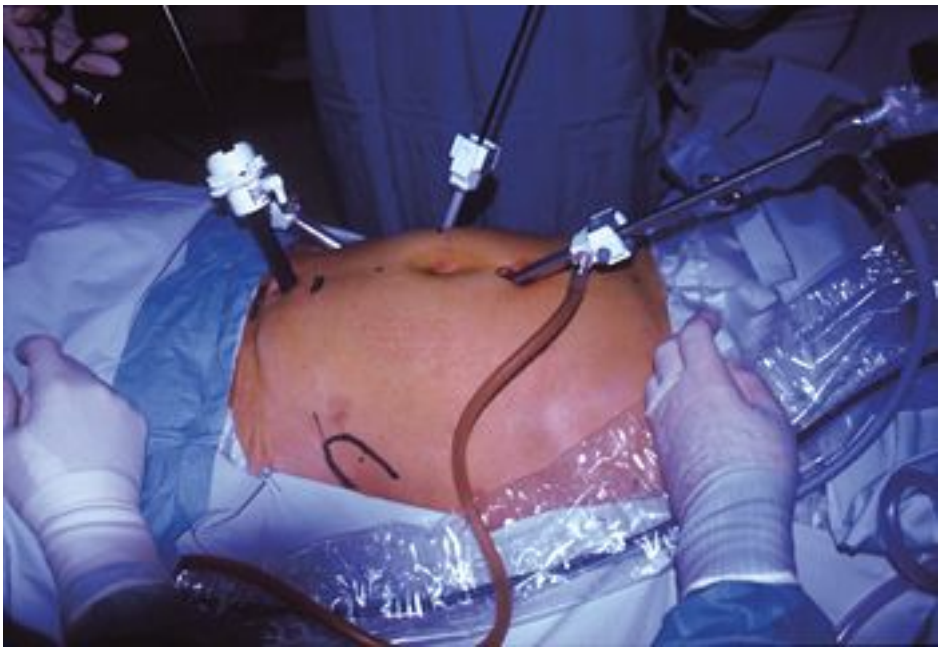

Fig. 5.6 – Opération d'endoscopie : Les trocarts placés places sur le corps humain.(http ://www.maitrise-orthop.com)

Plusieurs gestes sont réalisés par un chirurgien pendant une opération chirurgicale, nous citons les plus utilisés :

- **Palper :** Examiner, à travers l'instrument des structures anatomiques.
- **Faire une préhension :** Saisir un tissu sans produire de lésion apparente lors de la réalisation des gestes.
- **Biopsier :** Prélever un fragment de tissu par traction.
- **Couper :** Produire une section nette des tissus avec un instrument.
- **Suturer :** Coudre les lèvres d'une plaie ou de deux tissus d'un même organe ou de deux organes différents avec affrontement et pression adéquate.

En plus de la possibilité de reproduire des gestes médicaux, s'ajoutent aussi d'autres besoins tel un espace de travail suffisant pour balayer toute la cavité de l'opération chirurgicale. Ainsi la possibilité de créer le même environnement chirurgical par la disposition de plusieurs outils chirurgicaux dans la même opération.

Durant une opération réelle, le chirurgien est amené à changer d'outil, et pour que ce changement d'outil soit pris en compte dans l'environnement virtuel, l'insertion, donc la présence de l'outil ainsi que son type doivent être détectés automatiquement. Donc notre système doit être capable de détecter la présence de l'outil en premier lieu puis le type de l'outil inséré. Ce système devrait être capable de détecter les différents outils mis à disposition du chirurgien, qui seraient *a priori* au nombre de 10.

5.3.2 Analyse fonctionnelle d'une nouvelle interface

Nous pouvons déduire de notre analyse de besoin que le trocart actif doit permettre l'introduction et l'extraction sans heurt d'un instrument standard de chirurgie endoscopique non jetable dans un système simulant l'entrée d'un trocart. Il doit servir d'interface entre l'utilisateur muni de son trocart et l'environnement virtuel dans lequel il va évoluer. L'appareil devra donc restituer les efforts et contrôler la position du trocart. Les deux fonctions principales de ce système sont :

- Récupérer la position de l'outil puis la transmettre au monde virtuel où l'opération est simulée.
- Appliquer les forces que doit sentir le chirurgien, issues du monde virtuel.

Cet échange doit être assuré à une grande vitesse, à partir de 300Hz et pouvant aller jusqu'à 10KHz.

Pour pouvoir réaliser les gestes chirurgicaux décrits dans le paragraphe précédent, le système doit balayer toute la cavité de l'opération chirugicale et restituer l'effort à tout endroit de la cavité. Ce qui ce traduit par les 5 degrés de liberté comme le montre la figure 5.7. Ces 4 degrés de liberté sont les suivants :

- Translation
- Rotation propre
- Tangage
- Roulis
- Effort presseur

A l'extrémité utile (ie celle qui sert à opérer) de l'outil se trouve généralement une pince. La fermeture et l'ouverture de la pince sont commandées par une poignée. Le trocart

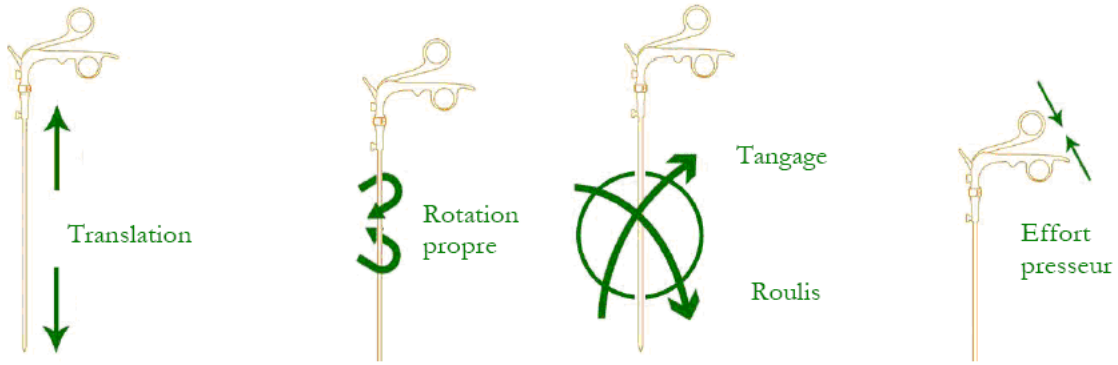

FIG. 5.7 – Les 5 degrés de liberté du trocart actif

actif doit pouvoir restituer les efforts au niveau de cette poignée, ainsi que déterminer la position de la pince. C'est le cinquième DDL "effort presseur".

Cette analyse profonde du système nous permet d'identifier les fonctions suivantes :

- Pouvoir accueillir sans heurt un instrument standard de chirurgie endoscopique de diamètre 5mm dans une structure d'accueil émulant un trocart.
- Identifier automatiquement l'instrument inséré.
- Détecter la présence ou l'absence de l'outil.
- Mesurer la position de l'instrument en translation.
- Mesurer les angles de rotation propre.
- Mesurer les angles de tangage et roulis.
- Restituer l'effort en tangage et roulis.
- Restituer l'effort en rotation.
- Restituer d'effort en translation.
- Mesurer l'ouverture et la fermeture de la pince.
- restituer les efforts au niveau de la poignée.

5.3.3 Contraintes techniques et réalisation du cahier des charges

Rappelons que la réalisation de notre système de retour d'effort se fait en partenariat avec un industriel. Donc en se servant du savoir faire de cet industriel dans le domaine de la simulation et surtout de l'industrialisation des produits nous avons pris en compte plusieurs aspects afin de définir un cahier des charges détaillé.

Architecture adoptée

L'architecture que nous avons proposée initialement est décrite par la figure 5.8. On distingue deux parties :

- Le coeur, dans lequel on introduit l'instrument,

– Le berceau, comprenant les structures mécaniques et électriques permettant la mesure des angles et la restitution des efforts de rotation.

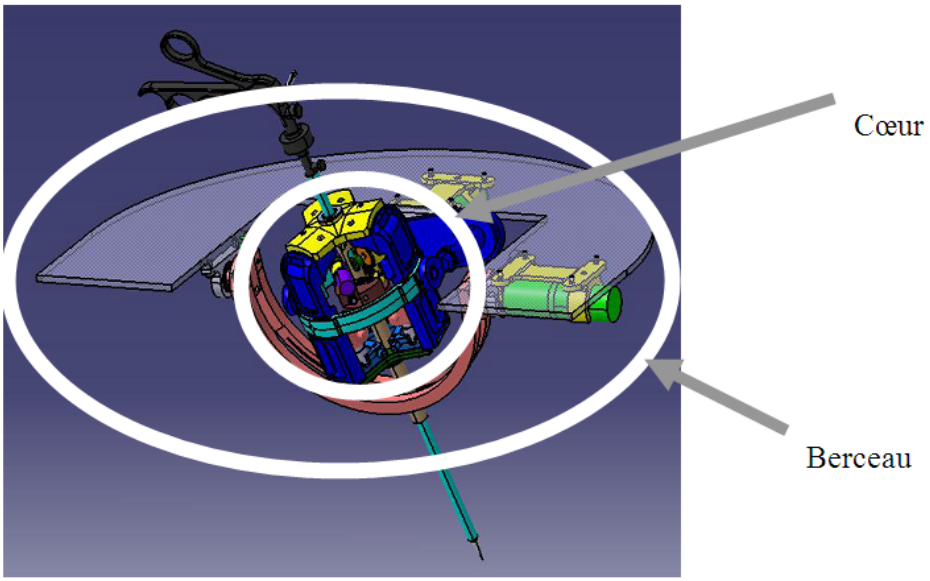

Fig. 5.8 – Coeur et berceau du trocart actif

L'ensemble doit être relié à un châssis de contrôle et d'alimentation, lui-même commandé par l'ordinateur de simulation.
L'ensemble du montage final aura alors l'allure montrée sur la figure 5.9.

Contraintes d'utilisation

Il faut pouvoir introduire ou extraire sans heurt un instrument standard de chirurgie endoscopique, tels que ceux présentés dans la figure 5.5. L'interface de manipulation de l'instrument se présente sous la forme d'un tube accueillant l'instrument. L'ensemble ne doit présenter aucun jeu parasite, ce qui implique un ajustement spécifique entre le corps de l'outil et le tube guide du trocart.

Il faut, dans la mesure du possible, ne pas être obligé de modifier l'instrument réel de façon irréversible pour, par exemple, pouvoir restituer l'effort en rotation Les frottements et les efforts d'inertie doivent être réduits au minimum, pour ne pas perturber la dynamique de l'outil ; le poids et les inerties doivent donc être réduits le plus possible. Le système doit être silencieux en fonctionnement. Des bruits mécaniques ne doivent pas troubler le chirurgien qui s'entraîne, pour s'approcher le plus possible des situations réelles.

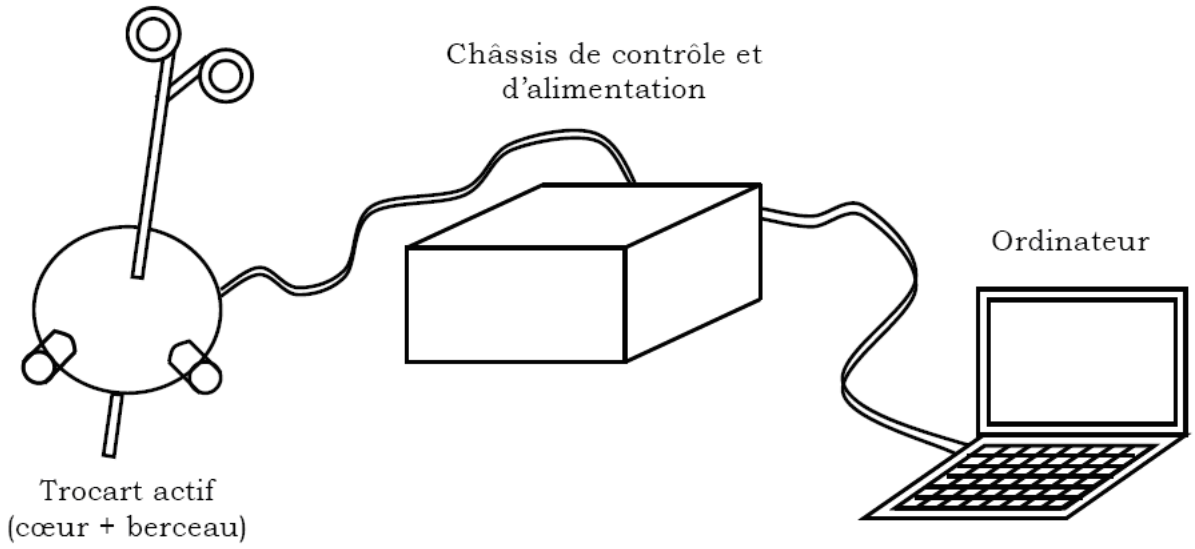

FIG. 5.9 – Schéma des différentes parties du simulateur

Encombrement

L'appareil doit être le plus compact possible pour que deux ou quatre trocarts puissent être rapprochés. Pour ce faire :

- Le coeur doit être compris dans une sphère de 80 mm de diamètre maximum
- La distance entre les axes de deux trocarts jointifs ne doit pas dépasser 100 mm

En outre, le coeur doit être réduit au maximum pour éviter les effets d'inertie. Par ailleurs, il doit être suffisamment léger et compact pour pouvoir être transportable par une personne.

Spécifications spatiales

Le trocart actif doit pouvoir accepter un corps d'instrument de 400 mm de long, et de 5mm de diamètre.
Justification : *le plus courant est de 300mm, mais il existe des corps d'instruments plus longs pour personnes obèses.*
L'outil doit pouvoir être bougé de plus ou moins 70 degrés autour des axes de roulis et de tangage. En pratique, la pointe de l'outil doit pouvoir atteindre n'importe quel point d'un espace conique de demi angle au sommet de 70 degrés et de hauteur 400 mm.
Observation : *Ces spécifications peuvent être partiellement réduites si elles entraînent des complexités ou des impossibilités mécaniques.*
La plage de variation de la rotation de l'instrument s'étend de 0 ° à 360 °

Cahier des charges

Pour la réalisation du cahier des charges, nous nous sommes basés sur [Tol95] qui analyse des actions élémentaires en chirurgie endoscopique. Les mesures de position doivent être effectuées avec une résolution suffisamment petite pour que le mouvement sur l'écran d'ordinateur paraisse fluide et continu. Les efforts d'inertie doivent être réduits au minimum. Quant aux frottements résiduels ils doivent être faibles, mais non nécessairement nuls. Notre système doit pouvoir appliquer des forces suffisantes pour restituer l'effort selon chaque degré de liberté.

5.4 Réalisation du dispositif

La réalisation d'une interface Homme/Machine dédiée à la simulation médicale comporte deux fonctionnalités majeures, le positionnement 3D et la restitution d'effort. Nous avons défini nos priorités pour réaliser ce système. En premier temps vient la fonction de détection de position de l'outil comme s'il s'agit d'un système de positionnement 3D. La restitution d'effort vient en second temps. Selon cet ordre : le retour d'effort selon le tangage et roulis suivi du retour d'effort de translation selon l'axe de l'outil et vient finalement la restitution d'effort de rotation autour de l'axe de l'outil.

5.4.1 L'architecture mécanique

Cette partie regroupe le coeur et le berceau (figure 5.8). Ces deux systèmes, cinématiquement liés doivent être considérés pour déterminer la cinématique du simulateur chirurgical et mener à bien la décomposition des efforts. Toute la partie mécanique a été modélisée sur un logiciel CAO CATIA.

Cœur

Le cœur permet l'introduction physique de l'outil, la restitution des efforts en translation selon la position de translation et la restitution des efforts en rotation propre selon l'angle de rotation.

Translation : L'outil est enfilé dans un premier fourreau carré (premier fourreau) comme le montre la figure 5.10. La forme de ce fourreau empêchera les rotations propres parasites entre le premier et le second fourreau. De plus, sa forme carrée permettra un meilleur appui des galets moteurs pour le retour d'effort en translation. La fixation de ce fourreau à l'outil est assurée par des joints aux extrémités du tube de l'outil. On enfile ensuite ce premier fourreau dans un second fourreau, d'intérieur carré et d'extérieur circulaire (second fourreau). C'est sur ce fourreau que viendra se fixer la couronne servant à l'appui des moteurs pour le retour d'effort en translation, et à celui en rotation propre. Le principe est le suivant : le premier fourreau coulisse parfaitement dans le second (figure 5.11). Pour générer un retour d'effort en translation, les moteurs exercent une rotation des galets qui, par frottement

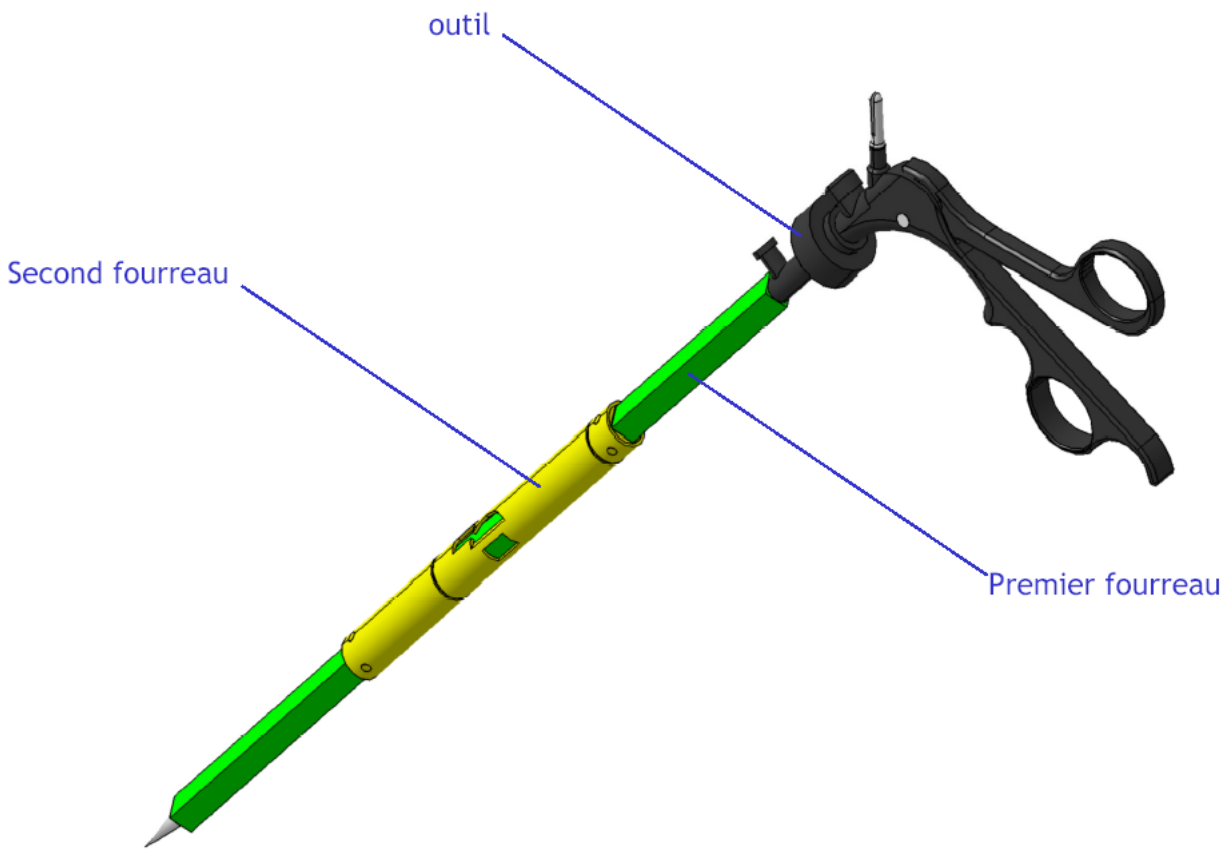

Fig. 5.10 – Habillage de l'outil pour la translation

contre le premier fourreau, créent le retour d'effort. L'appui des galets sur le premier fourreau est assuré par des ressorts présents dans le support moteur (jointure entre la couronne et le moteur), qui pressent effectivement les moteurs, et donc les galets, contre les fourreaux.

Rotation propre : Le retour d'effort en rotation propre est réalisé grâce à un engrenage. Cette solution n'offre certes pas une maîtrise « continue » du retour d'effort (nous sommes limités par la finesse de l'engrenage), mais offre l'avantage d'être très fiable et facile à mettre en place. Une roue dentée est fixée solidairement à la couronne. C'est sur cette roue dentée que vont travailler les moteurs générant le retour d'effort. Les moteurs réalisant le retour sont eux solidaires de la coque qui englobe tout le second fourreau, et autour de laquelle les fourreaux peuvent effectuer librement une rotation propre (figure 5.12). Lorsque l'utilisateur désire effectuer une rotation avec l'outil, c'est l'ensemble (premier fourreau + second fourreau + couronne + roue dentée) qui tourne, alors que la coque et les moteurs restent immobiles. Si un retour d'effort est commandé, les engrenages des moteurs vont tourner en sens inverse pour freiner la rotation imprimée par l'utilisateur.

Le berceau

Le berceau permet la restitution des efforts en tangage selon l'angle de tangage et la restitution des efforts en roulis selon l'angle de roulis.

Tangage - Roulis : Le retour d'effort en tangage et en roulis ainsi que le positionnement, sont tous les deux réalisés de manière découplée : l'utilisateur est ainsi libre de déplacer l'outil sans avoir à revenir à une position centrale pour changer entre tangage et roulis. Le tangage est réalisé en portant l'effort sur le bras de l'outil (en s'appuyant sur le premier fourreau). Un arceau entoure le bras, cet arceau est relié à des des moteurs situés sur la planche, permettant ainsi de réaliser le retour d'effort (figure 5.13). Le roulis est réalisé quant à lui au moyen d'un anneau en contact avec la coque, l'anneau étant mû par deux moteurs lorsqu'il y a commande en retour d'effort. Si le contact entre cet anneau et la coque était réalisé au niveau même des moteurs, il y aurait alors un problème, car lors du tangage la coque bougerait. C'est pourquoi le contact entre l'anneau et la coque est réalisé à 90° des moteurs, par deux vis qui peuvent tourner lors du tangage : en cas de tangage, l'anneau jaune (qui sert au roulis) ne tourne pas, mais la vis rose tourne. L'anneau orange est solidaire de la coque (figure 5.14). On constate donc que la fente de l'arceau de tangage permet d'exercer le roulis alors même que l'angle de tangage n'est pas nul, tout comme la vis de connexion.

5.4.2 Choix des composants

La performance de notre Simulateur à retour d'effort repose sur le principe suivant : repérer et envoyer à l'environnement virtuel la position de l'outil, puis appliquer à l'outil les forces calculées par l'environnement de la simulation. Elle est donc fortement dépendante de la performance du matériel.

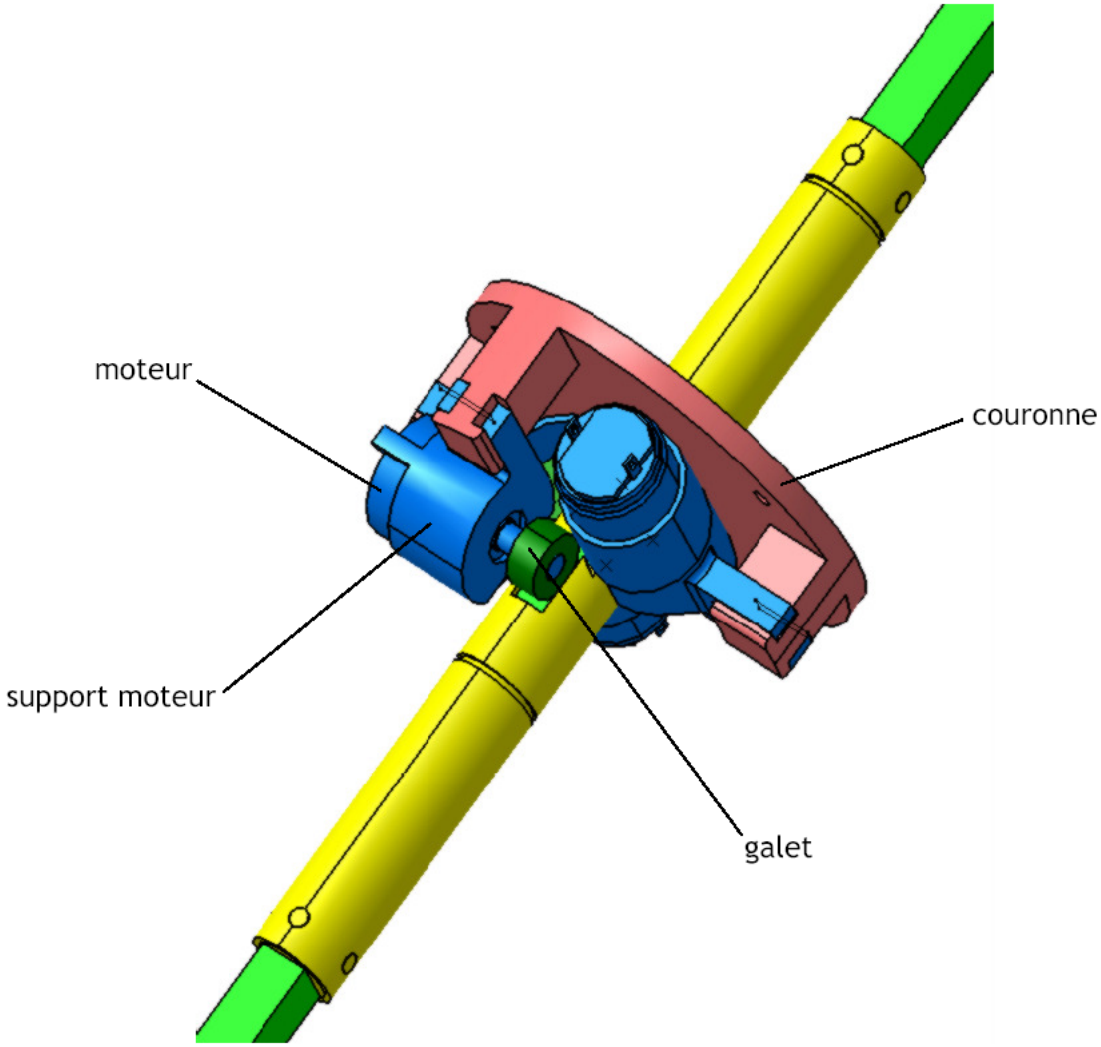

Fig. 5.11 – Principe de translation

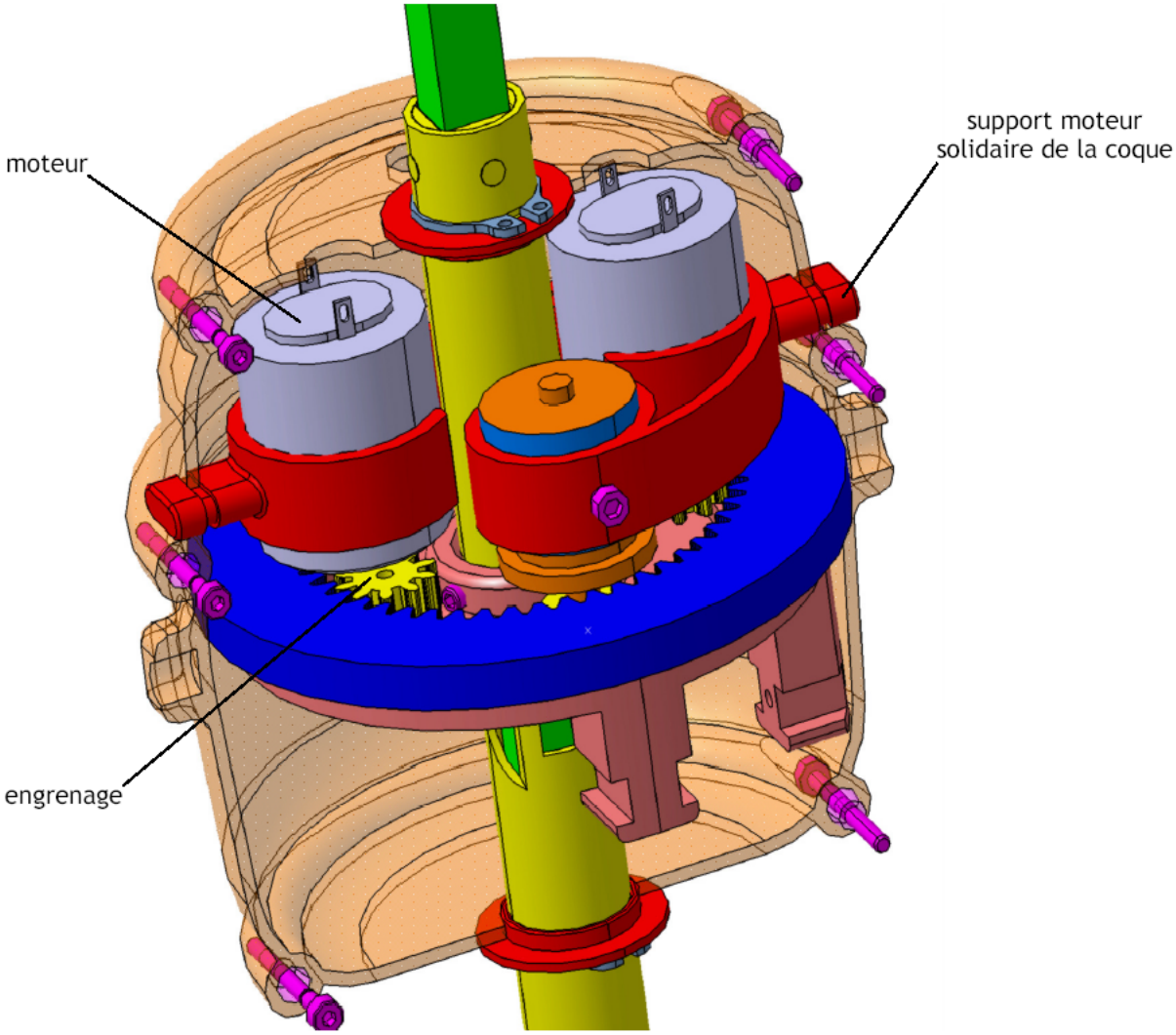

Fig. 5.12 – Le module de la rotation propre

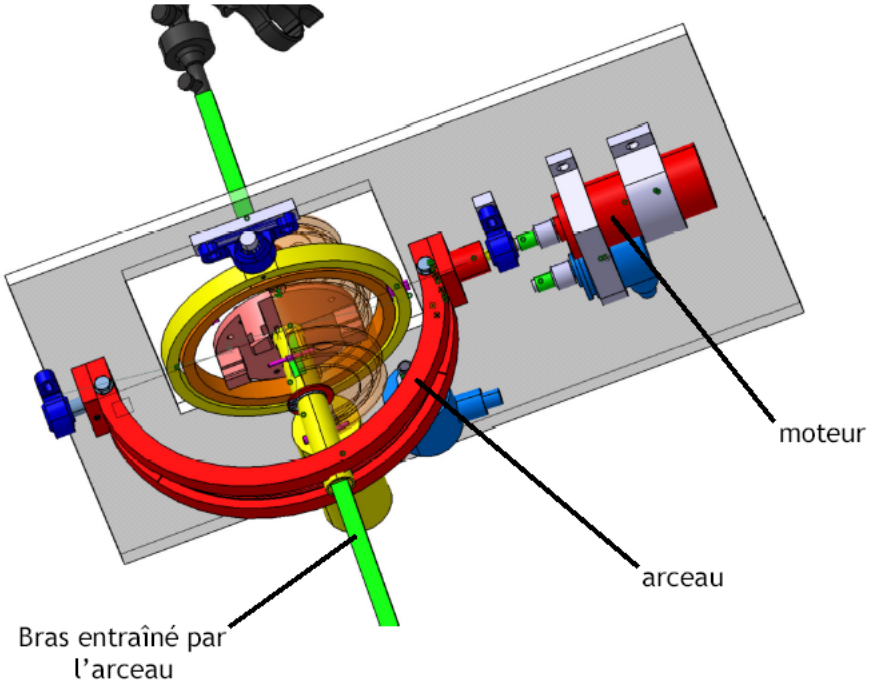

Fig. 5.13 – Le Berceau : tangage et roulis

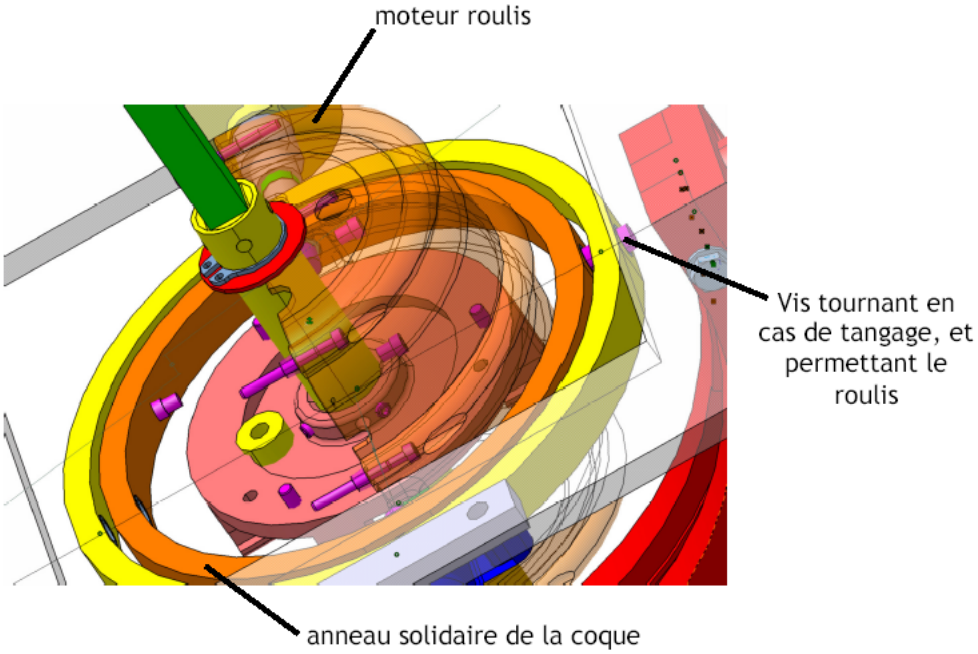

Fig. 5.14 – Dispositif du tangage

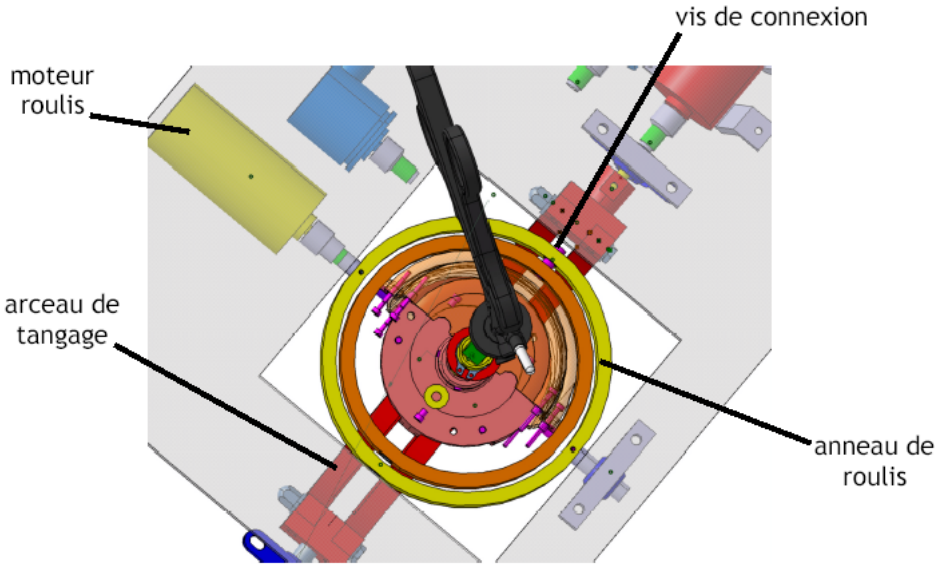

Fig. 5.15 – Dispositif du roulis

Le choix des Capteurs et Moteurs est alors une étape cruciale, puisqu'elle déterminera l'efficacité et la précision du repérage et de l'actionnement. Ces choix ont été faits, après avoir pris en compte l'architecture envisagée, les objectifs fixés et imposés par notre cahier des charges, et les considérations pratiques telles que le coût et l'encombrement.

Choix des actionneurs

Dans ce projet, on a opté pour l'utilisation des actionneurs électriques et plus spécifiquement de moteurs à courant continu. Les avantages vis-à-vis de l'application sont :

- Ils ont une bonne bande passante, ce qui permet la simulation des forces de hautes fréquences (difficile d'y parvenir avec des actionneurs à alliages à mémoire de forme et les actionneurs pneumatiques).
- Ils ont un comportement linéaire.
- Ils ne nécessitent pas d'équipements volumineux comme les actionneurs hydrauliques (alimentation d'huile en pression) et pneumatiques (compresseur).
- leur espace de travail n'est pas limité.

Le but est de trouver des actionneurs légers qui permettent d'avoir un couple maximum pour chaque DDL. Il y a trois critères :

- La minimisation de la masse des actionneurs.
- La minimisation de la taille des actionneurs pour réduire l'encombrement au maximum.
- La maximisation de l'effort produit par les actionneurs (dépendant de chaque DDL)

Après une recherche auprès de constructeurs des actionneurs, on a choisi les actionneurs Faulhaber qui sont des moteurs électriques puissants et avec peu de frottement.

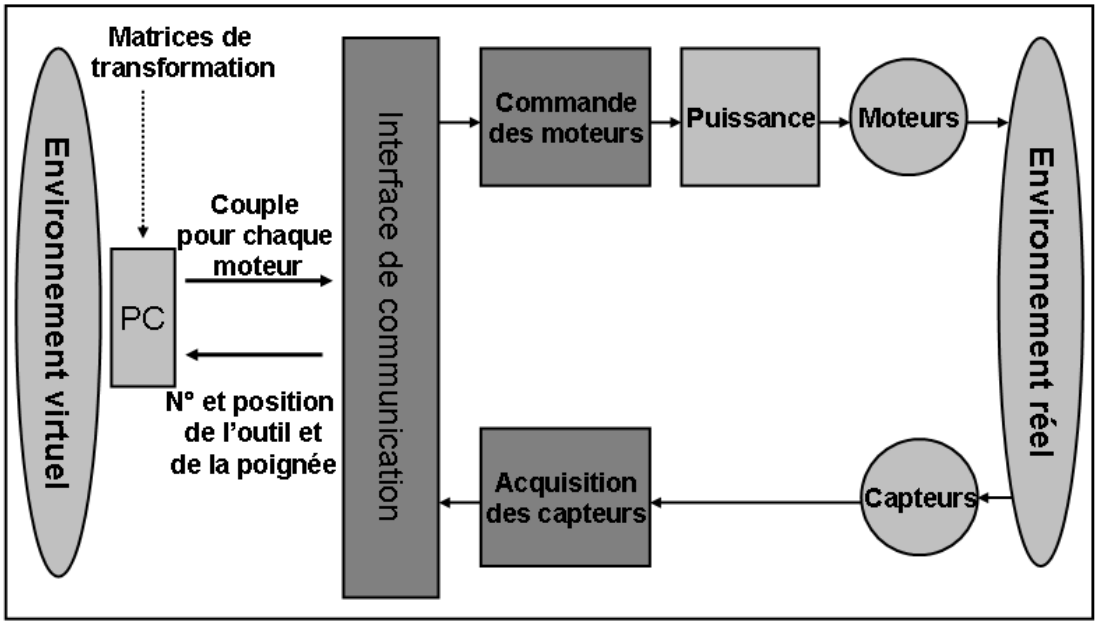

Fig. 5.16 – Architecture électronique de l'interface

Choix des capteurs

Le choix des capteurs a principalement été conditionné par les contraintes imposées pour la précision de la position de l'outil. En effet, le système doit pouvoir :

- Fonctionner à haute fréquence (1500Hz),
- Fournir une grande précision (rotation : 0.05° d'angle, translation : 0.05 mm).

Mais nous avons dû aussi tenir compte des compatibilités électroniques avec l'interface (analogique ou numérique), et de l'encombrement afin que les connexions n'entravent pas le mouvement.

D'autre part, il fallait aussi tenir compte du signal de sortie, et envisager un programme capable de le traiter. Notre choix s'est donc arrêté sur les capteurs à incrémentation, car ils répondent au mieux à nos exigences, et leurs signaux de sortie permettent un traitement facile afin de récupérer la position de l'outil.

5.4.3 Architecture électronique

Nous avons conçu une architecture électronique dont le schéma fonctionnel est représenté par la figure 5.16.

Ce schéma a été simplifié par l'introduction d'une carte de commande fournie par l'industriel partenaire du projet (figure 5.17). Cette carte remplace toutes les composantes qui s'occupaient de l'acquisition et du traitement de données. L'introduction de cet élément nous a permis de simplifier en grande partie les connexions et d'assurer la

Fig. 5.17 – Carte de commande fournie

rapidité et l'efficacité des calculs. La figure 5.18 nous montre la nouvelle architecture électronique basée sur la carte de l'industriel. On peut ainsi vérifier que la vitesse de réponse du système global répond aux exigences de «temps réel ». Cette carte fournit déjà à sa sortie des signaux continus qui n'ont donc plus qu'à être amplifiés. D'un autre côté, précisons simplement que la connexion entre les capteurs et la carte pour la capture de données se réalise directement : aucun élément intermédiaire d'interconnexion n'est nécessaire puisque la carte est déjà munie des résistances et des autres composants nécessaires pour assurer le bon fonctionnement global.

En intégrant le tout dans la maquette du simulateur chirurgical on peut arriver ainsi à détecter les variations de l'outil et à y répondre en exerçant le couple nécessaire par les moteurs.

5.4.4 Commande du trocart actif

La plupart des systèmes à retour d'effort sont des robots qui fonctionnent de façon inversée. Notre interface à retour d'effort n'a pas comme but l'exécution d'une tâche, mais la reproduction des forces appliquées par l'environnement virtuel sur l'utilisateur.

Il y a deux modes principaux pour contrôler une interface à retour d'effort détaillés dans [CC00] :

- **Le contrôle d'impédance**, dans ce mode de contrôle, on détecte le mouvement de l'utilisateur et on contrôle la force appliquée par l'interface.
- **le contrôle d'admittance**,on mesure les efforts appliqués par l'utilisateur à l'interface et on contrôle la position de l'interface.

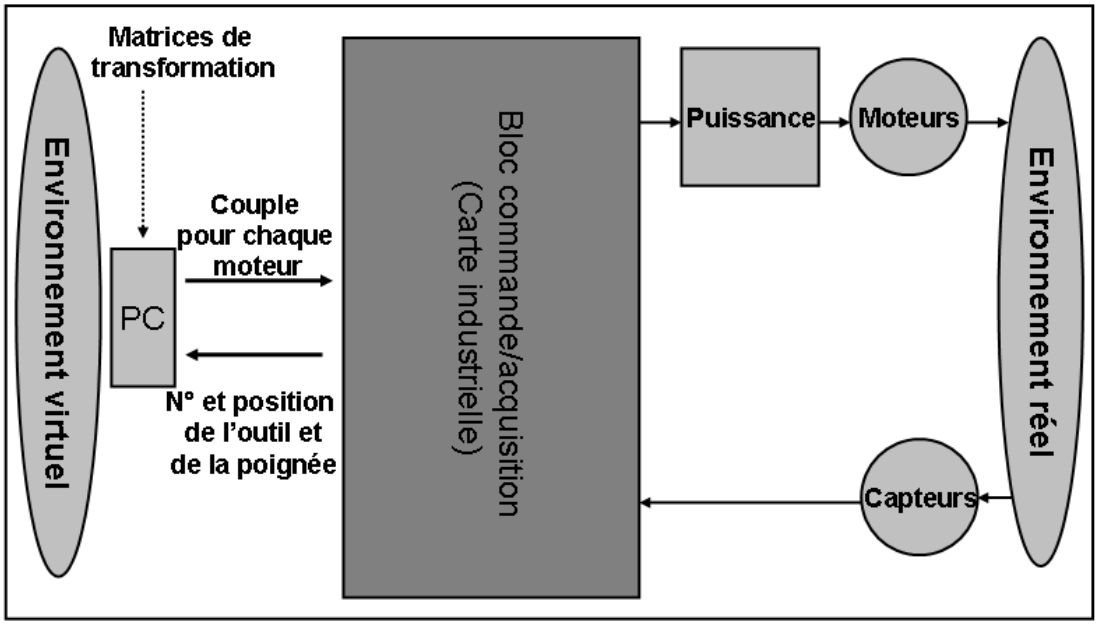

FIG. 5.18 – Nouvelle architecture électronique de l'interface

Pour notre système, nous avons opté pour la commande d'impédance, qui est plus pratique à utiliser avec l'environnement de simulation médicale. Cette méthode, utilisée en boucle ouverte (pas de correction de l'erreur entre les efforts souhaités et les efforts réellement appliqués) est satisfaisante quand l'interface a relativement peu de frottement et que ses masses en mouvement sont faibles.

Le principe du fonctionnement de notre système est basé sur, d'une part, la communication de l'interface des informations sur le mouvement de l'utilisateur (position de l'outil) à l'environnement virtuel en utilisant des capteurs, d'autre part, l'interface à retour d'effort communique les interactions de l'utilisateur avec l'environnement virtuel par la réaction des actionneurs. L'environnement virtuel nous donne la force virtuelle à appliquer en un point de l'outil afin de restituer les efforts ; cette force doit être décomposée et transposée en couples à appliquer aux différents moteurs de notre système.

Le but est donc de déterminer les couples à faire appliquer par les moteurs pour simuler les forces dues au contact de l'outil sur un organe. Dans les calculs qui suivent, on ne tiendra compte que d'une seule force, que l'on suppose ponctuelle, ce qui revient à dire que le torseur des actions correspondant au contact outil-organe, réduit au point de contact, ne comporte qu'une composante résultante. Il conviendra donc, dans le cas d'une force non ponctuelle, ou dans le cas de plusieurs forces appliquées à des endroits différents de l'outil de procéder à une sommation des résultats que nous allons établir ici. Notre environnement de simulation transmet au système les composantes de cette force en coordonnées cartésiennes ainsi que les coordonnées du point M où elle est appliquée. Dans toute la suite, on les notera respectivement (F_X, F_Y, F_Z) et (X, Y, Z). La figure

5.19 nous précise le schéma et les notations utilisées.

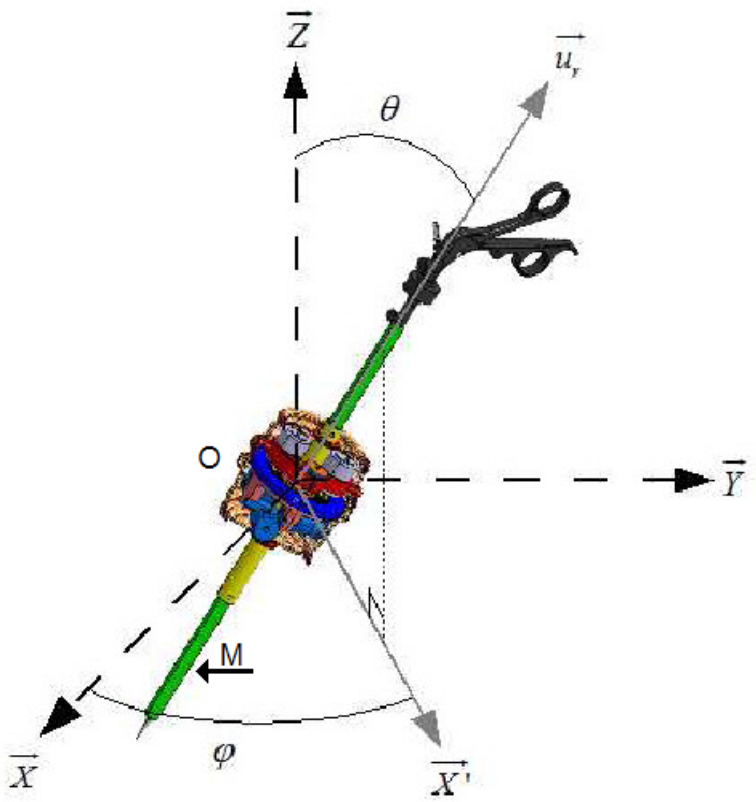

Fig. 5.19 – Schéma et notations

Calcul du moment en O

Calculons le moment de la force F en O, centre de rotation de l'outil :

$$\overrightarrow{\mathfrak{M}_o} = \overrightarrow{OM} \wedge \overrightarrow{F} = \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \wedge \begin{pmatrix} F_X \\ F_Y \\ F_Z \end{pmatrix} = \begin{pmatrix} Y.F_Z - Z.F_Y \\ Z.F_X - X.F_Z \\ X.F_Y - Y.F_X \end{pmatrix} \tag{5.1}$$

Nous avons donc ainsi l'expression des composantes des moments à appliquer selon $\overrightarrow{e_X}$,$\overrightarrow{e_Y}$ et $\overrightarrow{e_Z}$. Or nos moteurs sont selon les vecteurs $\overrightarrow{e_X}$,$\overrightarrow{e_Y}$ et $\overrightarrow{e_r}$, qui ne forment pas un trièdre orthogonal. Il va donc falloir tenir compte des composantes de $\overrightarrow{e_r}$. Nous avons pour cela besoin des matrices de passage entre les bases $\{\overrightarrow{e_X}, \overrightarrow{e_Y}, \overrightarrow{e_Z}\}$ et $\{\overrightarrow{e_r}, \overrightarrow{e_\theta}, \overrightarrow{e_\varphi}\}$

$$\begin{pmatrix} \overrightarrow{e_X} \\ \overrightarrow{e_Y} \\ \overrightarrow{e_Z} \end{pmatrix} = \begin{pmatrix} sin\theta.cos\varphi & sin\theta.sin\varphi & cos\theta \\ cos\theta.cos\varphi & cos\theta.sin\varphi & -sin\theta \\ -sin\varphi & cos\varphi & 0 \end{pmatrix} \cdot \begin{pmatrix} \overrightarrow{e_r} \\ \overrightarrow{e_\theta} \\ \overrightarrow{e_\varphi} \end{pmatrix} \tag{5.2}$$

$$\begin{pmatrix} \overrightarrow{e_r} \\ \overrightarrow{e_\theta} \\ \overrightarrow{e_\varphi} \end{pmatrix} = \begin{pmatrix} sin\theta.cos\varphi & cos\theta.cos\varphi & -sin\varphi \\ sin\theta.sin\varphi & cos\theta.sin\varphi & cos\varphi \\ cos\theta & -cos\theta & 0 \end{pmatrix} \cdot \begin{pmatrix} \overrightarrow{e_X} \\ \overrightarrow{e_Y} \\ \overrightarrow{e_Z} \end{pmatrix} \tag{5.3}$$

Où r, θ et φ sont donnés à chaque instant via les capteurs, puisqu'ils représentent les différentes rotations de l'outil par rapport au référentiel fixe $\{\overrightarrow{e_X}, \overrightarrow{e_Y}, \overrightarrow{e_Z}\}$.

Etant donné que les axes $\overrightarrow{e_X}$ et $\overrightarrow{e_Y}$ sont tous deux perpendiculaires à l'axe $\overrightarrow{e_Z}$, la seule façon de produire un moment selon l'axe $\overrightarrow{e_Z}$ est de le faire selon l'axe $\overrightarrow{e_r}$ et donc d'utiliser le moteur réservé à la rotation propre, puis de tenir compte des composantes engendrées selon $\overrightarrow{e_X}$ et $\overrightarrow{e_Y}$.

Grâce à la matrice de changement de base (Eq.5.2), un moment $\overrightarrow{\mathfrak{M}}$ selon $\overrightarrow{e_Z}$ a une composante $\frac{\mathfrak{M}}{cos\theta}$ selon $\overrightarrow{e_r}$. Pour produire un moment $\overrightarrow{\mathfrak{M}} = (X.F_Y - Y.F_X)\overrightarrow{e_Z}$, on va donc produire un moment $\overrightarrow{\mathfrak{M}} = \frac{X.F_Y - Y.F_X}{cos\theta}\overrightarrow{e_r}$. Ainsi, grâce aux matrices de changement de base, nous aurons le moment selon les deux repères :

$$\overrightarrow{\mathfrak{M}} = \frac{X.F_Y - Y.F_X}{cos\theta}\overrightarrow{e_r} \tag{5.4}$$

$$\overrightarrow{\mathfrak{M}} = \frac{X.F_Y - Y.F_X}{cos\theta}.(sin\theta.cos\varphi\overrightarrow{e_X} + sin\theta.sin\varphi\overrightarrow{e_Y}) + (X.F_Y - Y.F_X)\overrightarrow{e_Z} \tag{5.5}$$

Nous avons donc bien la composante souhaitée selon $\overrightarrow{e_Z}$. Reste à produire les composantes selon ($\overrightarrow{e_X}$ et $\overrightarrow{e_Y}$), en tenant compte des composantes déjà produites et que l'on vient d'exprimer. Nous allons donc soustraire les composantes selon $\overrightarrow{e_X}$ et $\overrightarrow{e_Y}$ données par l'équation 5.5 aux composantes selon ces mêmes vecteurs de l'équation 5.1.

On a donc :

$$\begin{cases} C_r = \frac{\mathfrak{M}_z}{cos\theta} = \frac{X.F_Y - Y.F_X}{cos\theta} \\ C_X = \mathfrak{M}_X - \mathfrak{M}_Z.cos\varphi.tan\theta = (Y.F_Z - Z.F_Y) - (X.F_Y - Y.F_X)cos\varphi tan\theta \\ C_Y = \mathfrak{M}_Y - \mathfrak{M}_Z.sin\varphi.tan\theta = (Z.F_X - X.F_Z) - (X.F_Y - Y.F_X)sin\varphi tan\theta \end{cases} \tag{5.6}$$

Ainsi, pour tenir compte du moment en O de la force $\overrightarrow{F}$, qui est à l'origine de la rotation de l'outil selon les différents axes, il suffit d'exercer un couple $C_r\overrightarrow{e_r} + C_X\overrightarrow{e_x} + C_Y\overrightarrow{e_Y}$ en O. Il suffit donc désormais de calculer la composante résultante, responsable du mouvement de translation de l'outil dans le fourreau.

Composante selon l'axe de translation

La seule translation possible de l'outil se fait selon l'axe $\overrightarrow{e_r}$, c'est à dire selon l'axe du fourreau. En effet, il n'est pas nécessaire de prendre en compte les autres composantes ($\overrightarrow{e_\theta}$ et $\overrightarrow{e_\varphi}$ puisque le fourreau empêche tout mouvement selon celles-ci. La matrice de changement de base 5.2 nous donne donc :

$$F_r = sin\theta.cos\varphi.F_X + sin\theta.sin\varphi.F_Y + cos\theta.F_Z \tag{5.7}$$

Calcul des couples moteurs

Désormais, nous connaissons les couples à appliquer selon les différents axes ainsi que la force nécessaire pour simuler la force de réaction normale (appelée ainsi car normale à la surface de contact) exercée par l'organe sur l'outil selon $\overrightarrow{e_r}$ Dans une première approximation, nous négligerons les forces de frottements dynamiques, en se plaçant dans le cadre de petits mouvements. De même, nous négligerons les frottements statiques, c'est à dire que nous supposerons les liaisons parfaites entre les différentes pièces mobiles de la structure. L'inertie de la structure a été réduite au maximum grâce notamment à l'emploi de matériaux composites. Nous ne cherchons qu'à retransmettre les efforts dus au contact outil/organe.

- Dans le cas de la **translation**, la transmission du mouvement se fait grâce au frottement du galet contre le fourreau de l'outil. Soit r_{gallet} le rayon du gallet, en supposant que nous avons un rendement $\eta_T \approx 0,7$ au niveau de contact avec le gallet, on a :

$$C_{moteurdetranslation} = \frac{r_{gallet}.(sin\theta.cos\varphi.F_X + sin\theta.sin\varphi.F_Y + cos\theta.F_Z)}{\eta_T} \tag{5.8}$$

- Dans la cas de la **rotation propre**, soit k le facteur de réduction de la roue dentée et η_R le rendement de l'engrenage qui est compris entre $0,95$ et $0,98$. On trouve :

$$C_{moteurderotation} = \frac{1}{k.\eta_R}\frac{X.F_Y - Y.F_X}{cos\theta} \tag{5.9}$$

- Dans le cas du **tangage et roulis**, La transmission est directe. Il n'y a donc pas de galet ou de roue dentée, et donc pas de facteur de réduction. On obtient donc les résultats suivants :

$$\begin{array}{l} C_{moteurdetangage} = (Y.F_Z - Z.F_Y) - (X.F_Y - Y.F_X)cos\varphi tan\theta \\ C_{moteurderoulis} = (Z.F_X - X.F_Z) - (X.F_Y - Y.F_X)sin\varphi tan\theta \end{array} \tag{5.10}$$

Les couples à appliquer sur les moteurs pour restituer l'effort sont issus d'un calcul simple, ce qui présente un avantage pour notre architecture. La commande du système sera simple et rapide et donc nous n'aurons pas besoin de l'implémenter dans un contrôleur de grande puissance de calcul. Précisons que nous agissons en boucle ouverte, c'est à dire sans correction entre les efforts de sortie souhaités et les efforts de sortie réels. Cette hypothèse est satisfaisante quand l'interface a relativement peu de frottements et que ses masses en mouvement sont faibles. Cette hypothèse correspond bien à notre application pour le médical parce qu'en réalité, le chirurgien ressent un frottement en déplaçant l'outil, ces frottements, qui sont dû au contact de l'outil avec le trocart et du trocart avec le corps du patient, seront assimilés à ceux de l'inertie du système mécanique.

5.4.5 Validation du prototype

La figure 5.20 nous illustre le modèle CAO du système complet. A la suite de la conception, Nous sommes passé à la fabriqcation du prototype.

Nous avons réalisé un prototype du système, la liaison avec l'ordinateur est assurée par une carte d'acquisition rapide via le bus firewire qui pourra réaliser un échange de données à une fréquence supérieure à 1000Hz. La figure 5.21 illustre cette réalisation.

Nous avons pu intégrer à ce prototype la fonctionnalité de récupérer la position de l'outil selon la translation, la rotation, le tangage et le roulis. Le positionnement et la réactivité du système étaient satisfaisants.

Quant à la restitution d'effort nous avons validé le retour d'effort selon le tangage et le roulis. Nous avons lié d'une façon simple ce prototype à notre modèle de tissu mous, ce qui nous a permis de valider qualitativement notre système de retour d'effort. Nous observons un bon suivi des moments et une bonne corrélation entre les déplacements observés à l'écran et les efforts transmis par le système.

Lors de notre conception, nous avons imposé des contraintes de tailles pour que plusieurs trocarts puissent être rapprochés, nous avons atteint cet objectif, mais ça cela a compliqué le câblage surtout pour la partie du cœur. Nous pouvons rapprocher jusqu'à quatre trocarts ce qui permet des simuler des opérations complexes. Lors d'une opération virtuelle, l'utilisateur sera amené à choisir l'outil chirurgical convenable pour l'insérer dans le trocart actif. Ce dernier, équipé par d'un lecteur de code barre, ou d'une puce RFID, identifiera l'outil et transmettra l'information à l'environnement virtuel.

En résumé, notre système a les caractéristiques suivantes :

-Diamètre du berceau	100 mm
-Rotation propre	0° - 360°
-Résolution en rotation propre	0.05°
-Couple maximal en rotation propre	0.2 N.m
-Tangage et roulis	+/-45°
-Résolution en tangage et roulis	0.01°
-Couple maximal en tangage et roulis	6 N.m
-Translation	300mm
-Résolution en translation	0,1 mm
-Retour d'effort maximal en translation	5 N
-Fréquence de travail	1000 Hz

5.5 Conclusion

Nous avons présenté dans ce chapitre nos travaux de conception et de réalisation du système de retour d'effort dédié à la chirurgie, le « trocart actif ». Notre système présente un avantage majeur sur l'existant, il peut être utilisé pour une grande variété d'opérations chirurgicales endoscopiques parce que nous l'avons conçu tout en s'intéressant au trocart, outil indispensable pour la chirurgies mini-invasive, et non pas à l'instrument chirurgical. Nous avons optimisé la conception de ce trocart afin de permettre l'utilisation de plusieurs trocarts actifs dans la même opération de chirurgie virtuelle et évidemment afin d'assurer le bon fonctionnement d'un système haptique pour garantir un bon rendement tactile. Avec ce système nous n'imposons pas le choix de l'outil chirurgical au chirurgien.

Nous avons développé ce système dans le cadre de la simulation médicale. L'utilisation de ce système ne se limite pas à ce genre d'application. Notre système sera très efficace et utile dans le cas des téléopérations. En téléopération à retour d'effort, l'opérateur réalise une tâche sur un environnement distant, en télé-manipulant un robot esclave[Zem05], via une interface haptique. L'opérateur peut ainsi ressentir et maîtriser les efforts que l'esclave applique sur l'environnement.

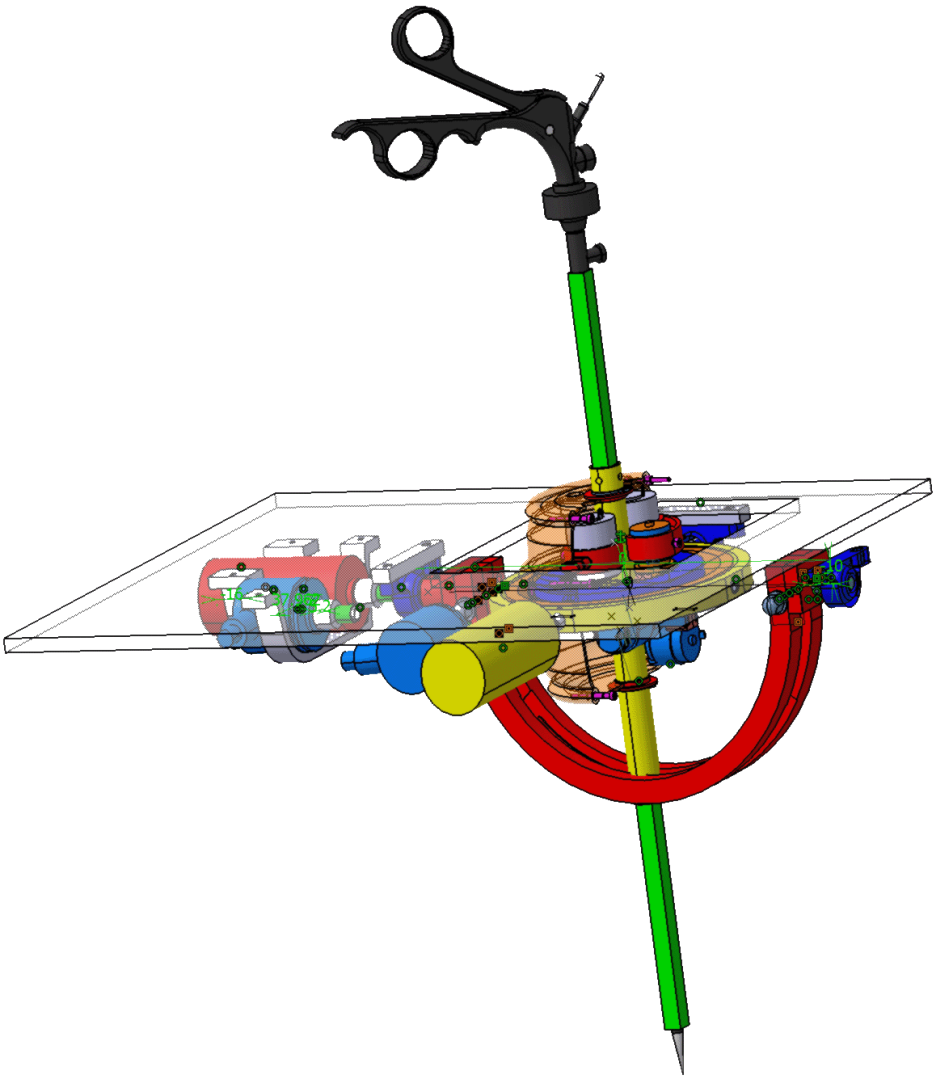

Fig. 5.20 – Le modèle CAO du système

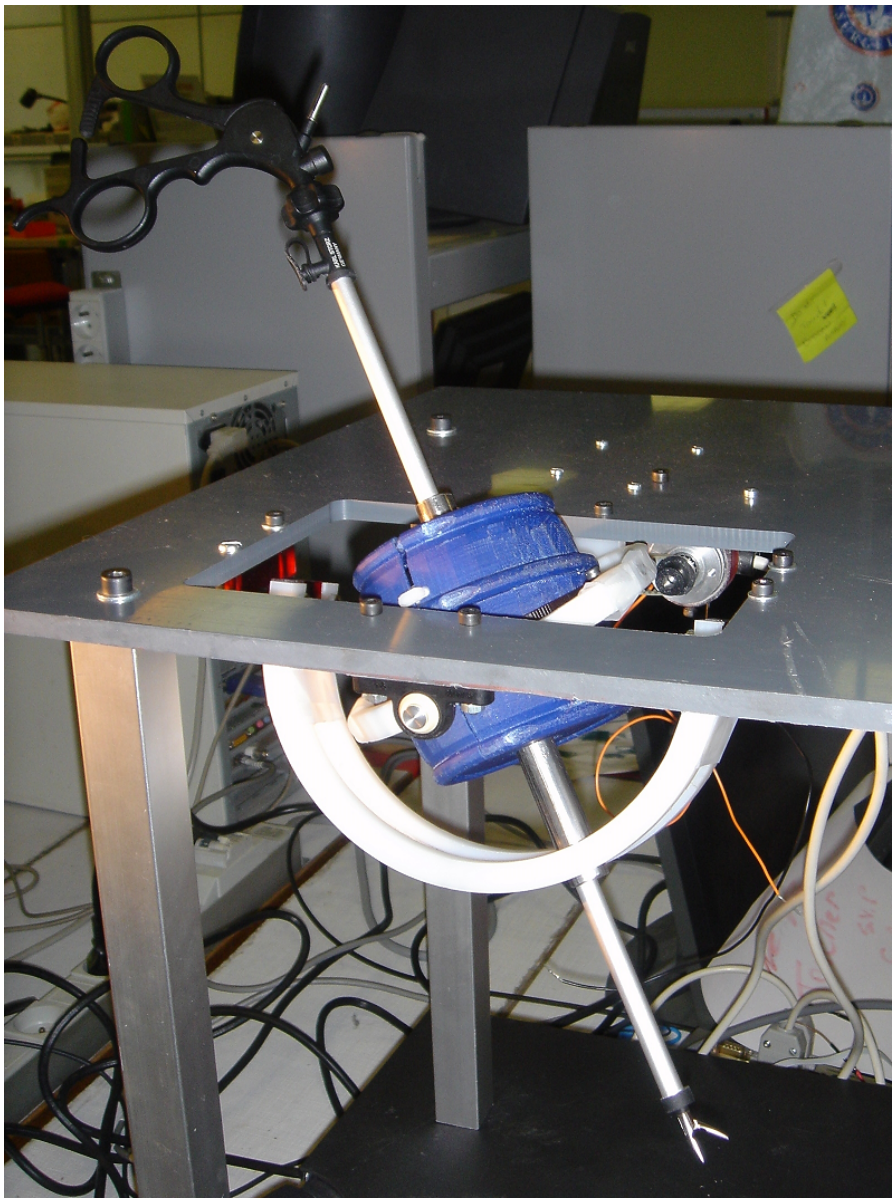

Fig. 5.21 – Le prototype réalisé

Conclusion Générale et Perspectives

Durant cette thèse, nous avons développé un modèle physique avec le meilleur compromis entre réalisme et rapidité de simulation. Ce modèle permet le changement de topologie. Cette implémentation a été basée sur une étude comparative des différents modèeles physiques existants. Il s'agit d'un modèle physique, issue de la Mécanique des Milieux Continus, qui offre la possibilité de paramétrer le modèle à partir de mesures expérimentales, et de construire une solution générique applicable à une grande variété de matériaux. C'est une synthèse entre deux présentations répandues : les masses-ressorts et les masses tenseurs. Elle se base sur le modèle des matériaux de Saint Venant Kirchhoff qui correspondent à une classe de matériaux hyperélastiques. Notre algorithme peut être étendu à différentes lois de comportement des matériaux.

Nous nous sommes ensuite intéressés à l'interaction avec/entre les organes. Nous avons divisé en particulier le problème en trois phases : la détection des collisions et des auto-collisions, la gestion de ces collisions et enfin la modification topologique de l'organe lors de la découpe.

Nous sommes allés plus loin pour la création d'un bon environnement de simulation médicale. Après une étude de dispositifs de retour d'effort existants, nous avons conçu et développé notre interface dédiée à la chirurgie virtuelle. Nous avons aussi réalisé une application simple, intégrant notre modèle déformable avec notre système de retour d'effort.

Enfin, il est important de noter que ces travaux se sont attachés à traiter l'intégralité d'un simulateur chirurgical : les tissus mous, les interactions (internes ou externes) et le système haptique dédié. Ce que nous considérons comme contribution en soi, facilitant la conception et la réalisation des systèmes de chirurgie virtuelle interactive.

Il reste toujours intéressant de développer autour de notre noyau de corps déformable et du système de retour d'effort, des applications variées pour la chirurgie endoscopique (gynécologie, urologie...) pour cela il faudrait développer une API complète pour la gestion du simulateur.

Notons que les applications de chirurgie virtuelle demeureront "'gourmandes"' en ressources informatiques, ce qui implique un souci constant d'optimisation pour assurer le rendu temps réel de la simulation d'une opération chirurgicale. Nous avons donc initié

des travaux de recherches sur ce thème. Ces travaux consistent à créer des unités de calculs implémentées sur des cartes FPGA permettant un calcul parallèle. Ces travaux ont abouti à cette publication[NAC07].

Pour une validation complète du système, il est nécessaire de récolter l'avis des spécialistes du domaine chirurgical sur le réalisme du rendu visuel et haptique.

Bibliographie

[Arn05] Antoine Arnail, Simulation chirurgicale : Détection et gestion des collisions et autocollisions, mémoire de stage de DEA. Ecole des Mines de Paris 2005.

[Bac05] George Baciu Wingo Sai-Keung Wong. Dynamic interaction between deformable surfaces and nonsmooth objects. In IEEE Transactions on Visualization and Computer Graphics, volume 11, pages 329-340, Washington, DC, USA, May 2005. IEEE Computer Society.

[Bar98] David Baraff and Andrew Witkin. Large steps in cloth simulation. In SIGGRAPH'98 : Proceedings of the 25th annual conference on Computer graphics and interactive techniques, pages 43-54, New York, NY, USA, 1998. ACM Press.

[BCG05] X.Brun, S. Chendeb, F. Goulette. Soft Tissue cutting adapted to Mass Spring Model. Proc. IEEE - Virtual Reality Int.Conf (VRIC), Laval, France, April 20-24, 2005

[BDLLG97] G. Buurdea, S. Deshpande, B. Liu, N. Langrana et D. Gomez. 1997, « A Virtual Reality-based System for Hand Diagnosis and Rehabilitaion »Presence - Teleoperation and Virtual Environments, MIT Press.

[Ber03] , Berkley J., Weghorst S., Gladstone H., Raugi G., Berg D. and Ganter M. « Banded Matrix Approach to Finite Element Modelling for Soft Tissue Simulation ». Virtual Reality : Research, Development and Applications, 4, 203-212, 2003.

[Bie00] D. Bielser - M.H. Gross, Interactive Simulation of Surgical Cuts, Proceedings of Pacific Graphics, 2000

[Bou00] D. Bourguignon and M.P. Cani. Controlling anisotropy in mass-spring systems. In Computer Animation and Simulation, pages 113-123, August 2000

[Br04] Xavier Brun, Simulation chirurgicale : découpe interactive de tissus mous avec un système à retour d'effort, mémoire de stage de DEA. Ecole des Mines de Paris 2004.

[Bri03] R. Bridson, S. Marino, and R. Fedkiw. Simulation of clothing with folds and wrinkles. In SCA'03 : Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation, pages 28-36, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association.

[Bro91] T.L. Brooks. "'Teleoperator Systeme Response for Nuclear Telepresence"' Proceeding of the fourth ANS Topical Meeting on Robotics and Remote Systemes. Ed Jamshidi Eicker, 1991 pp 547-556

[Bro98] M. Bro-Nielsen, Finite element modeling in surgery simulation. Proceedings of the IEEE Vol.86 p.490-503, 1998.

[Bro01] J. Brown, . Sorkin, C. Bruyns, J. Latombe, K. Montgomery, and M. Stephanides. Real-time Simulation of deformable objects ; Tools and application. In Computer Animation, Seoul, Korea, Novembre 2001.

[BW98] David Baraff and Andrew Witkin. Large steps in cloth simulation. In Michael Cohen, editor, SIGGRAPH'98 Conference Proceedings, Annual Conference Series. ACM SIGGRAPH, Addison Wesley, July 1998.

[Çak00] Çakmak, H. et Kühnapfel, U. Animation and Simulation Technique for VR-Training Systems in Endoscopic Surgery. Dans Eurographics Workshop on Animation and simulation' 2000 (EGCAS'2000), pages 173-185, Interlaken, Switzerland. 2000.

[CC00] Carigan C.R., Cleary K.R., Closed-loop force for haptic simulation of virtual environments, in Haptics-e, Vol. 1, No. 2,2000.

[CGC04] S. Chendeb, F. Goulette, P. Combet. Surface and Volume Organs Modelling Using Mass-Spring-Damper Systems for Surgical Simulation. Proc. IEEE - Virtual Reality Int.Conf. (VRIC)- Laval Virtual May 11-14, 2004, Laval, France.

[CGC04+] S. Chendeb, F. Goulette, P. Combet. Modélisation des tissus mous pour les simulateurs médicaux avec retour haptique. Journées des Jeunes Chercheurs en Robotique (JJCR 18), Ecole des Mines de Douai, 8-9 sept.2004.

[CG08] S. Chendeb, F. Goulette, Design and Realization of a Force Feedback System Dedicated to the Virtual Surgery. Medicine Meets Virtual Reality 16th. California junary 2008.

[Chev96] Mécanique des systèmes et des milieux déformables. Luc CHEVALIER. Ellipses 1996

[Cho67] Chou P. C., Pagano N. J. : Elasticity : Tensor, Dyadic, and Engineering Approaches . Van Nostrand, Princeton, 1967 (pp. 204-224).

[Cro92] M. Crouzeix, A.L. Mignot. Analyse numerique des équations différentielles. Masson, 1992.

[Cot00] S. Cotin - H. Delingette-N. Ayache, A Hybrid Elastic Model Allowing Real Time Cutting, Deformations and Force Feedback for Surgery Training and Simulation, Visual Computer Vol. 16 No8 p.437-452, 2000

[Cou83] Couarraze, G. Grossiord J.L. Initiation à la Rhéologie. Technique et Documentation-Lavoisier 1983.

[Dem96] J.P. Demailly. Analyse numérique et équations différentielles. PUG, 1996.

[Dev02] Olivier Devilliers and Philippe Guigue. Faster triangle-triangle intersection tests. Juin 2002. ftp ://ftp.inria.fr/INRIA/publication/publi-pdf/RR/RR-4488.pdf.

[DLB97] M. Dinsmore, N. Langrana et G. Burdea. 1997, « Virtual Reality Training Simulation for Palpation of Substance Tumors »IEE International Symposium on Virtual Reality and Application (VRAIS'97), Alburquerque, New Mexico.

[Ell96] R.E. Ellis, O.M. Ismaeil, M.G. Lipsett, "'Design and Avaluation of a High-Performance Haptic Interface"', Robotica, Vol. 4, 1996, pp. 321-327

[FLMC02] L. France, J. Lenoir, P. Meseure, C. Chaillou, « Simulation of a Minimally Invasive Surgery of Intestines », Proc. IEEE - Virtual Reality Int.Conf. (VRIC) Laval, 17-23 June 2002

[For03] C. Forest, Simulation de chirurgie par coelioscopie : contributions à l'étude de la découpe volumique, au retour d'effort et à la modélisation des vaisseaux sanguins, Thèse de l'école Polytechnique, 2003

[Fus06] Le traité de la réalité virtuelle, Vol. 2 L'interfaçage, l'immersion et l'interaction en environnement virtuel. Direction P.Fuchs. Les Presses, Écoles des Mines de Paris. Février 2006

[GC06] F. Goulette, S. Chendeb, Extended Mass-Spring (EMS) : a Framework for Fast Computation of Hyperelastic Materials in Real-Time Simulation of Surgery. Medical Image Computing and Computer Assisted Intervention (MICCAI), Copenhagen 2006

[Gel98] Gelder, A. Aproximate Simulation of Elastic Membranes by Triangulated Spring Meshes. In : Journal of Graphics Tools. v. 3, n. 2, pp. 21-42. (1998).

[Ger73] « Cours de mécanique des milieux continus »Tome 1 Théorie générale, Masson Paris PP 165-173, 1973.

[Gre95] Green, P., S., Jensen, J., F., Hill, J., W., Shah, A., « Mobile Telepresence Surgery », Acte de la conférence MRCAS'95, Baltimore, Maryland USA, (1995), p. 97-103.

[Hop02] Medical Image Computing and Computer Assisted Intervention- MICCAI 2002 Engineering Research Center for Computer-Integrated Surgical Systems and Technology. Johns Hopkins University, Baltimore, MD.

[Jim01] P. Jiménez, F. Thomas, and C. Torras. 3D Collision Detection : A Survey. Computers and Graphics, 25(2) : 269-285, April 2001. citeseer.ist.psu.edu/431815.html.

[JP99] D.Jmes and D.Pai Accurate real time deformable objects. Procccedings of Siggraph, pages 65-72, august 1999. cited on pages 8, 39, 40, 50, 54, 55

[kim97] Dong Jin Kim, Leonidas J. Guibas, and Sung Yong Shin. Fast collision detection among multiple moving spheres. In SCG'97 : Proceedings of the thirteenth annual symposium on Computational geometry, pages 373-375, New York, NY, USA, 1997. ACM Press.

[Kuh96] Kuhn, C., Kühnapfel, U., et Krumm, H.G. (1996). "Kalshruhe Endoscopic Surgery Trainer" A "Virtual Reality" based Training System for minimally Invasive Surgery". Dans Computer Assisted Surgery (CAR'96), Paris.

[Küh97] Kühnapfel, U., Kuhn, C., Hübner, M., Krumm, H.G., Maass ; H., et Neisius, B. (1997). " The Kalshruhe Endoscopic Surgery Trainer as an example for Virtual Reality in Medical Education". Minimally Invasive Therapy and Allied Technoloies (MITAI), 6 :122-125.

[Küh99] Kühnapfel, U., Çakmak, H., Maass ; H. (1999). "3D Modeling for Endoscopique Surgery". Dans IEEE Symposium on Simulation, pages 22-32, Delft University, Delt, NL.

[Kuh99] Kuhn, Ch. : M̈odellbildung und Echtzeitsimulation deformierbarer Objekte zur Entwicklung einer interaktiven Trainingsumgebung für die Minimal-Invasive Chirurgie;Doctoral Dissertation, University Karlsruhe (1999)

[Lai93] W.M. Lai, D. Rubin, E. Krempl « Introduction to continnum mechanics »Butterwort Heinemeann, Third Edition, 1993.

[Lan86] Landau L. D., Lifshitz E. M. : Theory of Elasticity . 3rd Edition, Pergamon Press, Oxford, 1986.

[Lar01] T. Larsson and T. Akenine-Mller. Collision detection for continuously deforming bodies, 2001. citeseer.ist.psu.edu/larsson01collision.html.

[Lau03] C. Mendoza - C. Laugier - O. Galizzi - F. Faure, Simulating Cutting in Surgery Applications using Haptics and Finite Element Models, Proceedings MICCAI, 2003.

[Lin98] M. C. Lin and S. Gottschalk. Collision detection between geometric models : a survey. Pages 37-56. citeseer.ist.psu.edu/lin98collision.html.

[Lyv03] Kévin Lyvan, Etude de matériaux élastiques définis par des barres. Application à la simulation en chirurgie endoscopique.Thèse de doctorat, université Paul Sabatier. Septembre 2003.

[Lau03] C. Mendoza - C. Laugier - O. Galizzi - F. Faure, Simulating Cutting in Surgery Applications using Haptics and Finite Element Models, Proceedings MICCAI, 2003

[MAA99] H. MaaB, U. Kühnapfel -Non invasive measurement of elastic properties of living tissue- Proc CARS99, Paris 23-26 june 1999, pp865-870.

[Mad95] Madhav Ponamgi, Dinesh Manocha, and Ming C. Lin. Incremental algorithms for collision detection between solid models. In SMA'95 : Proceedings of the third ACM symposium on Solid modeling and applications, pages 293-304, New York, NY, USA, 1995. ACM Press.

[Mar96] B. Marcus. Hands on : Haptic feedback in Surgical Simulation. In Medicine Meets Virtual Reality II : Interactive Technology & Healthcare : Visionary Applications for Simulation Visualisation Robotics. pp 134-139. San Diego, CA, USA : Aligned Management Associates 1996.

[Mar05] P.Martin, R.Jorge « Determination of material parameters for different hyperelastic models »VIII International Conference on Computational Plasticity, CIMNE, Barcelona, 2005

[Men03] C. Mendoza, Soft tissue interactive simulations for medical applications including 3D cutting and force feedback, Proceedings MICCAI, 2003

[Mes00] " Adeformable Body Model for Surgical Simulation " Journal of Visualization and Computer Animation, septembre 2000, p 197-208

[Mes02] P. Meseure, Habilitation à diriger des recherches, « Animation Basée sur la physique pour les environnements Interactifs temps réel »(in french) 2002.

[Mon01] C. Monserrat, V. Hermandez, M. Alcaniz, M.C. Juan, V. Grau. « A new approach for real time simulation of tissue deformations in surgery simulation »Computer Methods and Program in Biomedecine, pages 75-84, 2001.

[Mor01] A. Mor, Progressive Cutting with Minimal New Element Creation of Soft Tissue Models for Interactive Surgical Simulation, Thèse du Robotics Institute Carnegie Mellon University of Pittsburgh, 2001

[Moo88] Matthew Moore and Jane Wilhelms. Collision detection and response for computer animationr3. In SIGGRAPH '88 : Proceedings of the 15th annual conference on Computer graphics and interactive techniques, pages 289-298, New York, NY, USA, 1988. ACM Press.

[NAC07] M. Al Najjar, R. Ayoubi, S. Chendeb, H. Ziade, A Special Purpose Co-Processor Prototype for Virtual Surgery Simulators, medivis, pp. 54-59, CNF IEEE - International Conference on Medical Information Visualisation (MediViz 2007)

[Nie00] Nienhuys, H.W., and van der Stappen, A.F., « Combining Finite Element Deformation with Cutting for Surgery Simulations », Eurographics.00 (short presentations), Interlaken, 20-25 August 2000, 43-51.

[Nie01] H.W. Nienhuys - A.F. van der Stappen, A surgery simulation supporting cuts and finite element deformation, Proceedings Medical Image Computing and Computer Assisted Intervention (MICCAI), 2001

[Nie03] H.W. Nienhuys, Cutting in deformable objects, Thèse de l'Université d'Utrecht, 2003

[Pac05] Audrey Paccini, Developpement d'un modèle éléments finis 3D appliqué à la simulation d'opérations chirurgicales des tissus mous. Thèse de doctorat, Ecole des Mines de Paris. Novembre 2005.

[Pal02] Celine Paloc - Fernando Bello - Richard I. Kitney and Ara Darzi, Online multiresolution volumetric mass spring model for real time soft tissue de-formation, In Takeyoshi Dohi and Ron Kikinis, editors, MICCAI, 2002

[Pap02] Papadopoulos, E. ; Vlachos, K. ; Mitropoulos, D. ; Robotics and Automation, 2002. Proceedings. ICRA '02. IEEE International Conference on , Volume : 2 , 11-15 May 2002 Pages :2079 - 2084 vol.2

[Peq99] J.Péquignot, Optimisation géomètrique locale de maillages tétraèdriques, projet de fin d'étude de l'ESSI, INRIA, 1999

[PBB99] V. Popescu, G. Burdea et M. Bouzit 1999, « Virtual Reality Simulation Modeling for a Haptic Glove »Computer animation'99 Conference, pp. 195-200, Geneva, Switzerland.

[Pic02] G.Picinbono, H.Delingette, N.Ayache, Modèle déformable élastique non linéaire pour la simulation de chirurgie en temps réel. C.R.Biologies 325, pp, 335-344, 2002.

[Pro97] Xavier Provot. Animation Réaliste de Vêtements. Thèse de doctorat, Synthim - INRIA, France, December 1997.

[Rap98] M. Rappaz - M. Bellet - M. Deville, Modélisation numérique en science et génie des matériaux, Presse Polytechniques et Universitaires Romandes, 1998.

[Ryn02] Rynson W.H. Lau, Oliver Chan, Mo Luk, and Frederick W.B. Li. Large a collision detection framework for deformable objects. In VRST'02 : Proceedings of the ACM symposium on Virtual reality software and technology, pages 113-120, New York, NY, USA, 2002. ACM Press.

[Sal02] Mécanique des milieux continus Tome I, Tome II, Jean Salençon. Les éditions de l'écoles polytechnique 2002.

[Sat96] Richard Satava. The current status of the future. In Proc. OfAth Conf. Medicine Meets Virtual Reality (MMVR'96). Interactive Technology and the New Paradigm for Healthcare, pp 100-106. January 1996.

[Shi95] A touch reflexion system for interaction with remote and virtual environments. Dans Proceedings of IEEE/RSJ International Conference on Intelligent Robotics and Systems, Pittsburgh, PA (USA).

[Smi95] A. Smith, Y. Kitamura, H. Takemura, and F. Kishino. A simple and efficient method for accurate collision detection among deformable polyhedral objects in arbitrary motion. In VRAIS'95 : Proceedings of the Virtual Reality Annual International Symposium (VRAIS'95), page 136, Washington, DC, USA, 1995. IEEE Computer Society.

[Tay99] R.M. Taylor II. Août 1999, « Scientific Applications of Force FeedBack : Molecular Simulation and Microscope Control »- Course notes for 'Haptics : From Basic Principles to Advanced Applications' - SIGGRAPH 99 - Los Angeles, Californie, USA.

[Ter87] Terzopoulos D., Platt J., Barr A., Fleischer K. : Elastically deformable models. Proc. SIGGRAPH '87 (Anaheim, USA, 27-31 July 1987), 205-214.

[Tha91] D. Thalmann B. Lafleur, N. Magnenat Thalmann. Cloth animation with self-collision detection. In Proc. IFIP Conf. on Graphics Modeling, 1991.

[Tol95] Luis Toledo. "Analyse des actions élémentaires en chirurgie endoscopique - Applications au développement d'un instrument d'un instrument basé sur le concept de poignet articulé". Université René Descartes- Paris V 1995

[Ver97] E. Verron « Contribution expérimentale et numérique aux procédés de moulage par soufflage et de thermoformage »Thèse de Doctorat de Génie Mécanique, Ecole Centrale de Nantes, 1997.

[Vol95] Pascal Volino, Martin Courchesne, and Nadia Magnenat Thalmann. Versatile and efficient techniques for simulating cloth and other deformable objects. In SIGGRAPH'95 : Proceedings of the 22nd annual conference on Computer graphics and interactive techniques, pages 137-144, New York, NY, USA, 1995. ACM Press.

[Wag02] C. Wagner, N. Stylopoulos, R. Howe. The Role Of Force Feedback In Surgery : Analysis Of Blunt Dissection, The Tenth Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems. March 24-25, 2002 Orlando.

[WAR93] I.M. Ward, D.W. Hadley « An Introduction to the Mechanical properties of solid polymers »John Wiley & Sons Ltd, Chichster,1993

[Wel99] P.S. Wellman « Tactil Imaging », Thèse de doctorat, Division of Engineering and Applied Sciences. Cambridge, Harvard University, 1999.

[Zem05] Nabil ZEMITI : Rapport de thèse "<Commande en Effort des Systèmes Robotiques pour la Chirurgie Mini-Invasive"> UNIVERSITE PARIS 6. 05 Décembre 2005

[Zil95] C.B. Zilles and J.K. Salisbury, A constraint-based god-object method for haptic display, In Internationnal Conference on Intelligent Robots and Systems, pages 146-151, Pittsburgh, 1995.

Résumé

Le domaine médical voit apparaître depuis peu des simulateurs pédagogiques, à l'instar de ce qui existe depuis quelques années dans d'autres domaines (aviation, auto-école). En effet, la formation actuelle des chirurgiens se fait à la fois sur des cadavres (ce qui entraîne des difficultés grandissantes et des coûts élevés) et par compagnonnage (avec des chirurgiens expérimentés). Une forte demande est exprimée pour des simulateurs chirurgicaux pédagogiques, surtout dans le domaine de la chirurgie mini-invasive, avec une prise en compte des gestes et des sensations tactiles.

Ce travail de thèse, réalisé au Centre de Robotique (CAOR) de l'Ecole des Mines de Paris, couvre deux problématiques importantes pour la réalisation d'un simulateur chirurgical pédagogique : la modélisation temps réel des tissus mous et la réalisation d'un système haptique dédié à la chirurgie virtuelle.

Nous nous intéressons à la modélisation temps réel des organes déformables, nous mettons en œuvre une méthode physique, qui offre la possibilité de paramétrer le modèle à partir de mesures expérimentales, et de construire une solution générique applicable à une grande variété de matériaux. Il s'agit d'une approche issue de la Mécanique des Milieux Continus ; c'est une synthèse entre deux présentations répandues : les masses-ressorts et les masses tenseurs. Elle se base sur le modèle des matériaux de Saint Venant Kirchhoff qui correspondent à une classe de matériaux hyperélastiques. Une fois modélisé, l'organe virtuel doit interagir avec l'environnement, il s'agit de la détection des collisions (avec les outils) ou auto-collisions (l'organe se plie sur lui-même) ainsi que la gestion de ces collisions. Nous traitons aussi la découpe que peut subir un organe pendant l'opération.

Nous présenterons aussi nos travaux de conception et de réalisation du système de retour d'effort dédié à la chirurgie, le « trocart actif ». Notre système présente un avantage majeur sur l'existant, il peut être utilisé pour une grande variété d'opérations chirurgicales endoscopiques. Il s'agit de l'instrumentation du trocart, outil indispensable pour la chirurgie mini-invasive afin d'assurer au maximum que les outils de chirurgie ne heurtent pas la peau à l'endroit de l'incision. Nous l'appelons « trocart actif ». Nous avons optimisé la conception de ce trocart afin de permettre l'utilisation de plusieurs trocarts actifs dans la même opération de chirurgie virtuelle et évidemment afin d'assurer le bon fonctionnement d'un système haptique pour garantir un bon rendement tactile. Avec ce système nous n'imposons pas le choix de l'outil chirurgical au jeune chirurgien. Celui-ci aura le choix de l'outil chirurgical au cours de l'opération.

Abstract

The medical field is emerging recently some pedagogic simulators like what it exists from several years in other field like flight simulator or car driving simulator. Indeed, the current training of surgeons is performed on corpses (difficult, expensive and not practical) and companionship (with experienced surgeons). These difficulties involve a growing demand for surgical pedagogic simulators, especially in the field of mini-invasive surgery, with a consideration of gestures and tactile sensations.

This thesis work, conducted in the Robotics Center(CAOR) of the Ecole des Mines de Paris, covers two important issues for achieving a surgical simulator: the real time soft tissue modeling and the realization of a haptic system dedicated to the virtual surgery.

In order to model the deformable organ in real time, we conceive a physical method that offers the opportunity to parameterize our model starting from experimental measurements and to build a generic solution that can be applied to a wide variety of materials. Our approach consists of a synthesis between two kinds of presentation used in the field of continuum mechanics: the mass spring and the mass tensor. Our method is based on the Saint Venant Kirchhoff model which corresponds to a class of hyper-elastic materials. Once modelled, the body in ready to interact with the virtual environment. We, thus, propose some mechanisms for collisions detection with surgical tools and with the structure of the body itself. We propose also some mechanisms for collision management. A particular attention is given to the cutting mechanism that the body usually undergoes during surgery.

In the second section, we present our design and realization of a force feedback system dedicated to the virtual surgery. It is a new design of the trocar, an essential tool for the mini-invasive surgery. We call it "active trocar". It is a 5 DOF force feedback system. Our system has a major advantage on what exists. It can be used for a large variety of endoscopic surgical operations because we it is based on the trocar and not on the endoscopic surgery tool. We optimized the design of this trocar in order to allow the use of several devices in the same virtual operation and evidently in order to guarantee good tactile effects. With this system, the surgeon is free to choice the suitable surgical tool and fit into in the trocar.