



HAL
open science

Modélisation géométrique et dynamique d'un geste chirurgical

Eric Bainville

► **To cite this version:**

Eric Bainville. Modélisation géométrique et dynamique d'un geste chirurgical. Interface homme-machine [cs.HC]. Université Joseph-Fourier - Grenoble I, 1996. Français. NNT : . tel-00004351

HAL Id: tel-00004351

<https://theses.hal.science/tel-00004351>

Submitted on 28 Jan 2004

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.

THESE

Présentée par

Eric BAINVILLE

pour obtenir le titre de

Docteur de l'Université Joseph FOURIER – Grenoble I

(Arrêtés ministériels des 5.7.1984 et 30.3.1992)

Spécialité : Informatique

Modélisation géométrique et dynamique d'un geste chirurgical

Date de soutenance :

6 mars 1996

Composition du jury :

Nicholas AYACHE	Rapporteur
Francis SCHMITT	Rapporteur
Philippe CINQUIN	Directeur
Bernard LACOLLE	Examinateur
Roger MOHR	Examinateur
Claude PUECH	Examinateur
Roger SARRAZIN	Invité

Thèse préparée au sein du laboratoire :

TIMC-IMAG

Remerciements

Je remercie Jacques DEMONGEOT pour m'avoir accueilli dans le laboratoire TIMC pour sa sympathie et sa confiance.

Je remercie Philippe CINQUIN de m'avoir accueilli dans son équipe et d'avoir guidé mes recherches. Sa passion pour la recherche et son énergie m'ont soutenu infailliblement pendant ces années de thèse.

Merci également au professeur Roger SARRAZIN pour avoir soutenu avec enthousiasme la collaboration avec le service de chirurgie générale et thoracique de l'hôpital Michallon et en particulier avec Philippe CHAFFANJON.

Je remercie Roger MOHR de m'avoir fait l'honneur de présider mon jury de thèse et Nicholas AYACHE et Francis SCHMITT d'avoir été rapporteurs de ce mémoire ainsi que Claude PUECH et Bernard LACOLLE d'avoir accepté d'être membres du jury.

Pour avoir relu ce manuscrit et avoir contribué à son achèvement par leurs remarques, corrections et suggestions, je remercie Dominique ATTALI, Daniel BAINVILLE, Michel MÉO, Philippe CINQUIN et Nicholas AYACHE.

Je remercie les membres de l'équipe GMCAO pour tout ce qu'ils m'ont apporté ainsi que Maribel CHENIN pour sa gentillesse.

Merci en particulier aux "ours" (François ESTEVE, Olivier FRANÇOIS, Ali HAMEDH, François LEITNER, Bruno MAZIER, Pascal SAUTOT, Michel et Marian) pour leur amitié.

Enfin, pour leur soutien tout au long de ces années de recherches, avec les hauts et les bas que cela suppose, je remercie mes parents, mes frères Pierre-Yves et Marc, Fabrice DROUIN, Dominique ATTALI, sa sœur Carole et leur maman.

Grenoble le 22 avril 1996

Résumé

Ce travail traite du problème de la simulation et de l'assistance par ordinateur d'une opération chirurgicale. Nous abordons trois aspects de ce problème.

Tout d'abord nous décrivons un système d'assistance en temps-réel à une opération chirurgicale particulière : la rétroperitonéoscopie. Ce système permet d'afficher en continu des images de synthèse en fonction des déplacements de l'instrument chirurgical. Le chirurgien dispose ainsi d'informations supplémentaires lui permettant de rendre son geste plus rapide et précis. Nous détaillons la conception et la réalisation de ce système ainsi que les expérimentations sur spécimen anatomique.

Ensuite pour aller plus loin et simuler effectivement le comportement des organes du patient au cours de l'opération nous avons conçu un nouveau modèle de système de solides. Ce modèle fait cohabiter des solides rigides polygonaux et des solides fortement déformables représentés par un maillage de type "éléments finis". L'hypothèse d'une évolution quasi-statique du système et l'utilisation de lois de comportement élastique issues de la mécanique permettent d'obtenir un modèle robuste et réaliste. Nous détaillons l'implémentation de ce modèle et présentons quelques résultats proches des comportements "chirurgicaux".

Enfin nous étudions quelques outils mathématiques et algorithmiques utilisés dans les deux systèmes précédents : la représentation des rotations, la mécanique des milieux continus, la mise en correspondance rigide de nuages de points appariés et la détection d'objets circulaires et elliptiques et de grilles dans des images en niveaux de gris.

Abstract

Our work deals with the computer simulation and assistance of a surgical intervention. Three aspects of this problem are studied.

First we describe a system of real time assistance to a surgical operation : the retroperitoneoscopy. This system displays synthesized images depending on the position of the surgeon's tool in the patient's body. Therefore the surgeon has new informations at his disposal helping him to increase the precision and efficiency of his gesture. The design and the realization of the system are explained and experiments on anatomic specimens are described.

Next in order to simulate the behaviour of the organs involved in the surgical intervention we propose a new physically based model of a system of solids. This model mixes polygonal rigid bodies and highly deformable smooth objects. The deformable objects are represented by finite element meshes. A robust and realistic model is obtained under the hypothesis of a quasi-static evolution and through the use of hyperelasticity. We explain in detail the model and give results of experiments.

At last we study some mathematical and algorithmic tools used in the previous models : the representations of rotations, the mechanics of deformable solids, the rigid matching of two set of points and the detection of circular features and grids of points in grayscale images.

Mots-clés

Assistance informatique d'un geste chirurgical — Simulation d'un geste chirurgical — Animation de solides déformables — Représentation des rotations — Mise en correspondance rigide — Segmentation globale de structures connues

Table des matières

Introduction	13
I Simulation et assistance d'un geste chirurgical	19
1 Introduction	21
2 Assistance et simulation de la rétro-péritonéoscopie	23
2.1 Rétro-péritonéoscopie et médiastinoscopie	24
2.2 Matériel et système	27
2.2.1 Repérage du médiastinoscope par rapport à l'image 3D . . .	27
2.2.2 Quelles images? sous quelle forme?	31
2.2.3 Système d'assistance proposé	32
2.3 Protocole	34
2.3.1 Phase pré-opératoire	35
2.3.2 Calibrage du médiastinoscope	35
2.3.3 Recalage du patient	35
2.3.4 Navigation assistée	36
2.4 Expérimentation	36
2.5 Conclusion	37
3 Conclusion	39
II Animation de systèmes de solides	41
1 Introduction	43
2 Animation et modelage par ordinateur	45
2.1 Définitions	45
2.2 Déformations et géométrie	46
2.2.1 Solides rigides articulés	46
2.2.2 Systèmes de particules	47
2.2.3 Surfaces et volumes déformables	48
2.3 Contraintes	50
2.3.1 Ordre: géométrique/cinématique/dynamique	50
2.3.2 Portée: fixe/conditionnelle	50

2.3.3	Formulation : forte/faible	50
2.3.4	Interactivité	51
2.3.5	Globalité	51
2.3.6	Exemples de contraintes	51
2.4	Contrôle	53
2.4.1	Contrôle pour le modelage	53
2.4.2	Contrôle pour l'animation	54
2.5	Conclusion	55
3	Un modèle pour la simulation d'un geste chirurgical	57
3.1	Contexte et idées	57
3.2	Géométrie et déformations	58
3.2.1	Fonctionnalités	59
3.2.2	Maillages de type éléments finis	61
3.2.3	Solides rigides polyédriques	69
3.3	Contraintes	70
3.3.1	Liaisons	70
3.3.2	Contacts	71
3.3.3	Lois de comportement élastique	72
3.4	Contrôle	73
3.4.1	Détection des interpénétrations	73
3.4.2	Optimisation sous contraintes	75
3.5	Résultats	76
3.6	Conclusion	81
4	Conclusion	83
III Outils		85
1	Introduction	87
2	Représentation des rotations	89
2.1	Définitions	89
2.1.1	Rotations	90
2.1.2	Définitions annexes	90
2.2	Propriétés	92
2.2.1	Sous-espaces globalement invariants	92
2.2.2	Produit de réflexions	93
2.2.3	Exponentielle d'un endomorphisme antisymétrique	93
2.2.4	Matrice orthogonale	94
2.3	Représentations en dimension 2	94
2.3.1	Angle	94
2.3.2	Matrice antisymétrique	94
2.3.3	Complexe unitaire	95
2.4	Représentations en dimension 3	95
2.4.1	Axe et angle	95

2.4.2	Vecteur rotation	96
2.4.3	Matrice antisymétrique	96
2.4.4	Quaternion unitaire	96
2.4.5	Angles d'Euler	97
2.5	Passage d'une représentation à l'autre en dimension 3	97
2.5.1	$M \rightarrow$ vecteur	98
2.5.2	$M \rightarrow q$	99
2.5.3	$q \rightarrow$ vecteur rotation	100
2.5.4	$q \rightarrow M$	100
2.5.5	Vecteur rotation $\rightarrow M$	100
2.5.6	Vecteur rotation $\rightarrow q$	100
2.6	Les rotations en dimension 4	101
2.6.1	Introduction	101
2.6.2	Polynôme caractéristique	101
2.6.3	Cas $\Delta = 0$	102
2.6.4	Cas $\Delta > 0$	105
2.7	Conclusion	106
3	Modélisation du comportement des solides déformables	107
3.1	Mécanique des milieux continus	107
3.1.1	Déformation	108
3.1.2	Etude locale des déformations	108
3.1.3	Lois mécaniques d'équilibre	110
3.2	Comportement des solides	112
3.2.1	Nécessité d'une loi de comportement	112
3.2.2	Lois usuelles	112
3.2.3	Comportements limites	113
3.3	Discretisation et résolution	114
4	Mise en correspondance	115
4.1	Position du problème	116
4.2	Mise en correspondance de nuages de points appariés	116
4.2.1	Mise en équations	116
4.2.2	Problème linéaire associé	117
4.2.3	Grandeurs associées au problème	118
4.2.4	Solutions utilisant la représentation matricielle	119
4.2.5	Solutions utilisant les quaternions unitaires	120
4.2.6	Solutions utilisant le vecteur rotation	122
4.3	Conclusion	124
5	Segmentation	127
5.1	Détection grossière d'éléments circulaires et elliptiques	129
5.1.1	Approches existantes	129
5.1.2	Solution proposée	130
5.2	Localisation de la projection d'une grille régulière d'éléments circulaires	134
5.2.1	Signature	134

5.2.2	Corrélation angulaire	134
5.3	Conclusion	138
	Conclusion	139
	Bibliographie	141

Table des figures

1	Domaines couverts par notre problème	14
2	Plan de ce document et dépendances entre les chapitres	15
3	Opération sous contrôle vidéo	23
4	Lomboscopie : voies d'abord et geste	25
5	Rétropéritonéoscopie iliaque : voies d'abord et geste	25
6	Médiastinoscopie	26
7	Instruments utilisés pour la rétropéritonéoscopie	26
8	La position des organes du patient est connue dans une image tridimensionnelle	27
9	Repères et transformations intervenant dans le calcul de la transformation de \mathcal{M} vers \mathcal{C}	28
10	Le localisateur tridimensionnel Optotrak	28
11	Le marqueur en forme de boule à 24 diodes et son repère de calibrage	29
12	La géométrie du médiastinoscope est connue dans le repère \mathcal{M}	30
13	Le médiastinoscope sur son socle de calibrage	31
14	Images affichées en cours d'opération	33
15	Les différents types de processus et de données du système	34
16	Echographe muni de marqueurs	36
17	Fonctionnalités d'un objet associé à un solide	60
18	Élément paramétrique quadratique ; solide "anneau" constitué de cinq éléments de ce type	62
19	Les trois systèmes de coordonnées associés à un élément	63
20	Fonctions de pondération des éléments quadratiques à 9 nœuds	63
21	Problèmes de recollement pouvant apparaître entre des frontières de degrés différents	64
22	Champs de gradient dans l'anneau sous plusieurs déformations	65
23	Points d'évaluation de la fonction dans les méthodes de Gauss à $4\Gamma 9$ et 16 points	66
24	Densité d'énergie élastique dans l'anneau déformé	67
25	Arc de courbe constituant les contours des solides	68
26	Echantillons des contours de l'objet "clé"	68
27	Grandeurs associées à un curseur	69
28	Curseur généré par une interpénétration solide/demi-espace	72
29	Curseurs générés par une interpénétration solide/solide	73

30	<i>Quadtree</i> formé à partir d'échantillons de la frontière des solides . . .	75
31	Détail de l'arbre montrant le fonctionnement de l'algorithme de détection de contacts	76
32	T rigide droit au dessus d'un plan horizontal	77
33	T rigide incliné au dessus d'un plan horizontal	77
34	Etoile rigide reposant sur un plan horizontal	78
35	Deux T rigides posés l'un sur l'autre sur un plan horizontal	78
36	Deux T rigides entre deux parois qui se rapprochent	79
37	Anneau élastique reposant sur un plan horizontal	79
38	Anneau élastique reposant à l'intérieur d'un autre anneau	80
39	T rigide posé sur un anneau élastique	80
40	5 solides élastiques dans une cavité	81
41	Déformation d'un objet	108
42	Contraintes internes en un point	110
43	Nos 16 images de test	128
44	Régions intérieure I et extérieure E du masque	131
45	Centres détectés sur nos 16 images de test	133
46	Signatures obtenues à partir des images des séries C et D	135
47	Angles déduits des signatures de la figure précédente	136
48	Angles obtenus selon la méthode de corrélation angulaire	137

Introduction

Contexte

Depuis la fin des années 1980 on constate l'introduction de l'informatique et de la robotique dans certaines phases des gestes chirurgicaux. Ces techniques sont utilisées par exemple en neurochirurgie (ponction précise de tumeurs) en ophtalmologie (simulation d'opérations au laser) en chirurgie orthopédique (positionnement optimal de prothèses et d'implants). L'outil informatique permet d'accroître la qualité et l'efficacité du geste.

Nos recherches se placent dans ce contexte en abordant le cas de la chirurgie endoscopique et en particulier la rétro-péritonéoscopie et la médiastinoscopie. L'objectif de nos travaux est de réaliser un **système d'assistance et de simulation** de ces gestes chirurgicaux.

La rétro-péritonéoscopie consiste à introduire un endoscope rigide derrière le péritoine afin de prélever des échantillons ganglionnaires au voisinage des vertèbres lombaires. Ce geste est présenté dans la partie *Simulation et assistance d'un geste chirurgical*. Sur le même principe la médiastinoscopie concerne les ganglions de la partie inférieure du médiastin.

La recherche dans le domaine des gestes médicaux et chirurgicaux assistés par ordinateur (GMCAO) s'effectue sur deux niveaux. D'une part elle vise à **intégrer des composants** (modèles, outils et données) issus de plusieurs domaines comme l'illustre la figure 1 afin de réaliser et d'expérimenter des prototypes cliniques. D'autre part elle étudie les **fondements mathématiques et algorithmiques** de ces composants; ce travail de fond indispensable permet d'améliorer l'efficacité des outils existants et d'ouvrir de nouvelles perspectives d'applications.

En ce qui concerne notre objectif il faut d'une part modéliser l'ensemble des objets constituant la zone d'intervention (organes, instruments) et d'autre part créer les interfaces, les instruments spécifiques, les outils mathématiques et algorithmiques d'assistance et de simulation.

Plan

Nos recherches se sont orientées dans trois directions complémentaires et liées :

- Réaliser et expérimenter un système d'assistance visuel à la rétro-péritonéoscopie.

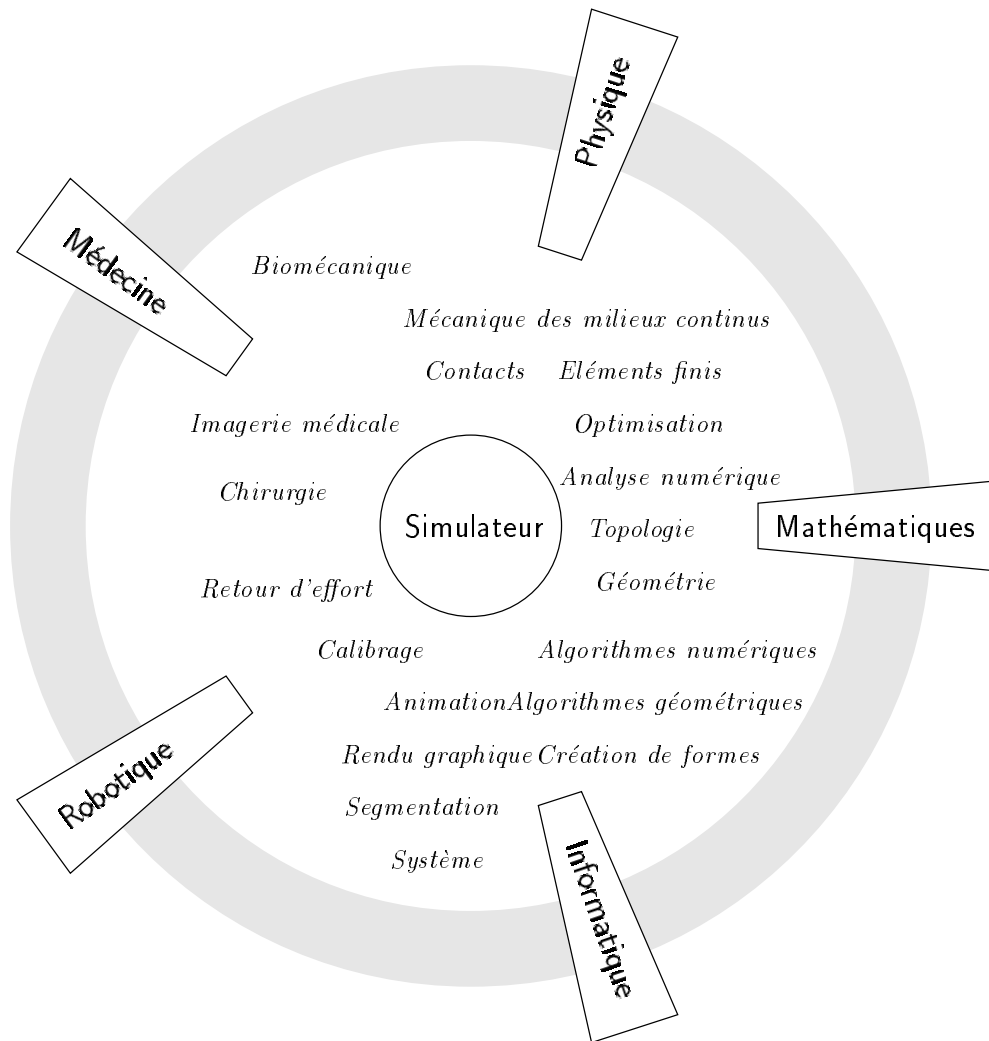


Fig. 1 - *Domaines couverts par notre problème.*

- Modéliser de façon réaliste le comportement du système de solides de comportements variés (rigides ou mous, mobiles ou fixes) formé par les organes, les tissus et les instruments du chirurgien.
- Etudier les différents problèmes mathématiques et algorithmiques posés par ces deux points.

Ce document comporte trois parties correspondant à ces directions. La figure 2 résume le plan de ce document et montre les relations entre les différents chapitres.

Partie I

La réalisation d'un système d'assistance visuelle à la rétro-péritonéoscopie est l'objet de la première partie: *Simulation et assistance d'un geste chirurgical*. Ce système en est à la phase des tests cliniques et est utilisé également pour d'autres opérations.

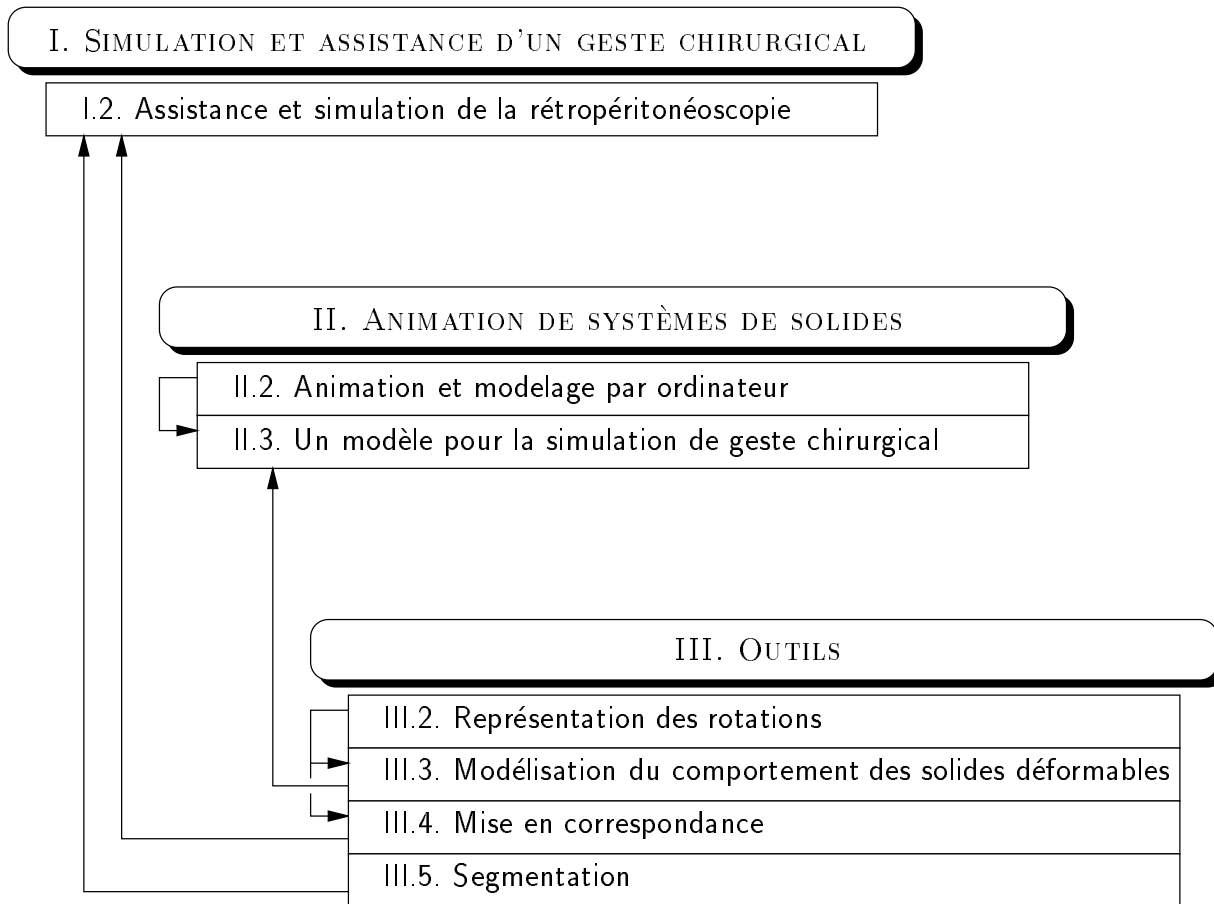


Fig. 2 - Plan de ce document et dépendances entre les chapitres. Une flèche $A \rightarrow B$ indique que B utilise des résultats de A .

Nous y décrivons la rétroperitonéoscopie puis l'architecture logicielle adoptée et enfin les expériences réalisées pour valider le système. On identifiera également les problèmes de calibrage, segmentation, mise en correspondance que l'on retrouve dans tous les systèmes de GMCAO.

Partie II

Dans la deuxième partie *Animation de systèmes de solides*, nous étudions tout d'abord les systèmes existants d'animation et de modelage de formes basés sur les lois de la physique (chapitre 2). Dans cette étude, nous séparerons les trois composantes essentielles de ces systèmes : la représentation de la géométrie et des déformations, les contraintes et l'algorithme de contrôle.

Ensuite, nous définissons les conditions que doit remplir un système d'animation pour pouvoir être utilisé dans un simulateur de geste chirurgical et nous proposons un **système original simulant l'évolution quasi statique d'un ensemble de solides rigides et déformables** (chapitre 3).

L'utilisation d'éléments finis pour représenter la géométrie et les déformations des

solides permet d'atteindre un réalisme physique satisfaisant comme l'illustrent les résultats présentés. Notre modèle traite efficacement le problème de la **cohabitation d'objets très déformables et d'objets rigides** en permettant l'utilisation simultanée de plusieurs représentations des solides dans le même système.

L'hypothèse d'évolution quasi statique n'est pas incompatible avec la simulation de geste chirurgical et permet de simplifier le traitement des contacts permanents entre surfaces rigides et déformables.

Partie III

La troisième partie *Outils* comprend quatre études indépendantes. Cette partie essentiellement bibliographique aurait pu constituer une annexe. Néanmoins sa taille et le fait que certains des résultats présentés soient originaux nous ont conduits à la conserver dans le corps du document.

La **représentation des rotations** étudiée dans le chapitre 2 tient une place importante dans les algorithmes de mise en correspondance et de calibrage. Nous rappelons quelques définitions et propriétés puis donnons des résultats spécifiques aux dimensions usuelles 2 et 3 avec quelques contributions ponctuelles. Une section originale porte finalement sur les rotations en dimension 4 dont nous pensons qu'elles peuvent intervenir dans les problèmes de vision.

Dans le chapitre 3 purement bibliographique nous rappelons les équations de la **mécanique des milieux continus** et les lois de **comportement des solides déformables**.

Le problème de la mise en correspondance omniprésent dans le domaine des GMCAO est présenté au chapitre 4 à travers sa forme la plus simple la **mise en correspondance de nuages de points appariés**. Dans ce problème on souhaite déplacer un ensemble de points pour le superposer à un autre en sachant quel point d'un ensemble correspond à chaque point de l'autre.

Les algorithmes de mise en correspondance servent dans les GMCAO d'une part lors du calibrage des différents capteurs de position et d'autre part lors du recalage du patient par rapport à un référentiel connu du bloc opératoire.

Nous exposons les différents algorithmes existants et présentons une idée de solution utilisant la représentation des rotations par leur "vecteur rotation".

A travers la localisation précise d'images de billes métalliques dans des images tridimensionnelles (obtenues par scanner) ou bidimensionnelles (radiographies, images vidéo) nous abordons au chapitre 5 le problème de la segmentation.

Nous avons utilisé des billes métalliques pour servir de points de repère anatomique dans la mise en correspondance entre des images scanner et le spécimen anatomique lors des premières expérimentations du système d'assistance décrit dans la première partie.

Nous voyons comment on peut **localiser grossièrement** un ensemble d'objets circulaires ou elliptiques dans une image puis comment **raffiner ce résultat** à des résolutions inférieures au pixel.

Nous exposons enfin dans ce même chapitre quelques résultats sur la **détection automatique globale** de grille d'objets circulairesΓutilisées pour le calibrage de systèmes de vision. Au lieu de détecter précisément tous les objets circulaires de l'image et de construire une grille à partir de ces pointsΓnous proposons de détecter avec précision l'ensemble des objets constituant la grille en une seule fois.

Première partie

**Simulation et assistance d'un
geste chirurgical**

Chapitre 1

Introduction

Le domaine des gestes médicaux et chirurgicaux assistés par ordinateur (GMCAO) se situe à l'interface entre l'informatique, la robotique et la médecine.

Les applications des GMCAO sont multiples. Tout d'abord, la précision du geste chirurgical est augmentée, ce qui améliore la sécurité et les performances de la technique. Ensuite, le chirurgien a la possibilité de planifier son geste et de s'entraîner sur un simulateur avant de réaliser effectivement l'opération. De façon similaire, la mesure a posteriori de la précision du geste fournit au chirurgien un retour facilitant son apprentissage. Enfin, ces techniques de planification, réalisation et évaluation de gestes réels peuvent être utilisées uniquement dans leur partie simulation pour permettre au chirurgien d'enseigner la méthode opératoire.

Les techniques mises en oeuvre dans ce domaine peuvent être rassemblées en trois thèmes : la perception, le raisonnement et l'action.

La **perception** concerne l'acquisition d'informations, utilisant des appareils d'imagerie médicale (amplificateurs de brillance, tomographie, résonance magnétique, échographie...) ou des capteurs de position et de surfaces (vidéo, mécaniques, optiques, électro-magnétiques). Le traitement de bas niveau de ces images, comme l'extraction des contours, l'extraction de formes, le calcul de déplacements à partir des positions de points caractéristiques... fait aussi partie de la perception.

Le **raisonnement** concerne l'analyse et la synthèse de ces informations, soit automatiquement par le système (par exemple pour le point d'insertion du ligament de remplacement sur la surface des os du genou), soit par le chirurgien (par exemple pour la trajectoire d'insertion des vis dans le pédicule d'une vertèbre). Le raisonnement permet au chirurgien de connaître avec précision le geste à effectuer.

La phase d'**action** consiste à aider le chirurgien à réaliser ce geste, en ajoutant des informations visuelles, sonores et gestuelles à sa perception habituelle.

La recherche dans ce domaine s'étend sur une large palette de problèmes (voir l'introduction de ce document). La plupart de ces problèmes ne sont pas spécifiques aux GMCAO. Les recherches menées dans le cadre des GMCAO peuvent ainsi trouver des applications dans les autres domaines, et réciproquement, les GMCAO peuvent bénéficier des avancées réalisées dans ces domaines.

Dans cette partie, nous décrivons la rétroperitonéoscopie, qui est un exemple

d'opération de chirurgie endoscopique. Ensuite nous décrivons un système d'assistance à la rétroperitonéoscopie mis au point et expérimenté au cours de cette thèse en collaboration avec l'équipe de chirurgie générale et thoracique de l'hôpital de Grenoble. Ce système est à mettre en parallèle avec les techniques actuelles de tomodensitométrie et d'imagerie par résonance magnétique interventionnelles : dans notre cas on transporte "virtuellement" l'image scanner ou IRM dans la salle d'opération.

Cette description sera l'occasion d'identifier les problèmes qui se posent lors de la réalisation d'un tel système. L'étude de ces problèmes fait l'objet des deux autres parties de ce document.

Chapitre 2

Assistance et simulation de la rétro-péritonéoscopie

Les techniques de chirurgie endoscopique se développent rapidement. Elles consistent à atteindre l'organe à opérer en introduisant les micro-instruments dans le corps du patient par des incisions de petite taille (voir l'introduction de la partie *minimal access surgery* dans [102]).



Fig. 3 - *Opération sous contrôle vidéo (prof. Brichon).*

L'opération s'effectue alors généralement sous contrôle vidéo (figure 3) : une caméra manipulée par l'assistant suit en permanence l'extrémité des instruments du chirurgien ; ce dernier opère en regardant uniquement l'image fournie par la caméra. Dans ce cas, du gaz est généralement injecté entre les organes pour ménager un espace permettant les déplacements de la caméra et des outils.

Dans d'autres cas, comme ceux de la rétro-péritonéoscopie et de la médiastinoscopie, le chirurgien déplace d'une main un "tube" dénué d'optique – le médiastinoscope – dans la cavité abdominale, en manipulant de l'autre main les instruments placés à l'intérieur du canal opérateur que constitue le médiastinoscope. L'opération s'effectue sans injection de gaz, le médiastinoscope servant lui-même à refouler ou

comprimer les organes environnants.

La rétroperitonoscopie et la médiastinoscopie présentent de fait des avantages pour le patient : le choc opératoire et les risques d'infection sont réduits et diminuant ainsi le temps d'hospitalisation et d'indisponibilité.

Cette technique présente cependant quelques inconvénients pratiques. Tout d'abord en cas de problème (hémorragie l'opération se révélant plus délicate que prévu...) il peut s'avérer nécessaire de convertir l'opération en une laparotomie (ouverture large des parois) et de terminer l'opération avec les techniques classiques. D'autre part la réalisation du geste réclame une plus grande expérience de la part du chirurgien. En effet il doit en permanence se représenter mentalement la position de son instrument par rapport aux organes nobles environnants et à la zone qu'il doit atteindre qui sont hors de son champ de vision réduit à une zone de 2 cm de diamètre à l'extrémité du médiastinoscope.

L'apprentissage de la rétroperitonoscopie et de la médiastinoscopie est difficile puisque seul le chirurgien manipulant le médiastinoscope voit les organes se trouvant à l'extrémité du tube : il n'est pas possible au maître de guider ou corriger le geste de son élève.

Ces raisons ont motivé l'étude et la conception d'un système informatique de simulation et d'assistance applicable à différentes opérations de chirurgie endoscopique et en particulier à la rétroperitonoscopie.

Un tel système peut être utilisé pour **préparer** un geste réel avant l'opération pour **simuler** un geste fictif servant à l'apprentissage ou pour **assister** le geste en cours d'opération.

L'utilité d'un système d'assistance per-opératoire apparaît surtout dans les cas suivants :

- la cible est petite (5 millimètres de diamètre)
- l'individu est obèse et la cible est entourée de zones graisseuses importantes
- la cible est placée au voisinage de zones dangereuses (voies sanguines ou urinaires, plèvres et œsophage pour la médiastinoscopie).

Dans ce chapitre nous présentons la rétroperitonoscopie et la médiastinoscopie. Ensuite nous décrivons le matériel et le système mis au point au cours de nos expériences puis le protocole opératoire retenu. Nous exposons finalement les différentes expériences effectuées afin de valider notre système.

La description du système sera l'occasion d'identifier un certain nombre de problèmes. Nous reviendrons en détail sur certains de ces problèmes dans la partie *Outils*.

2.1 Rétroperitonoscopie et médiastinoscopie

La rétroperitonoscopie et la médiastinoscopie (figure 6) sont des opérations similaires visant respectivement à atteindre les chaînes ganglionnaires abdominales et médiastinales. On effectue deux types de rétroperitonoscopie : lomboscopie (figure

4) ou rétropéritonéoscopie iliaque (figure 5) selon que l'exploration s'effectue dans la portion lombaire ou iliaque du rétropéritoine.

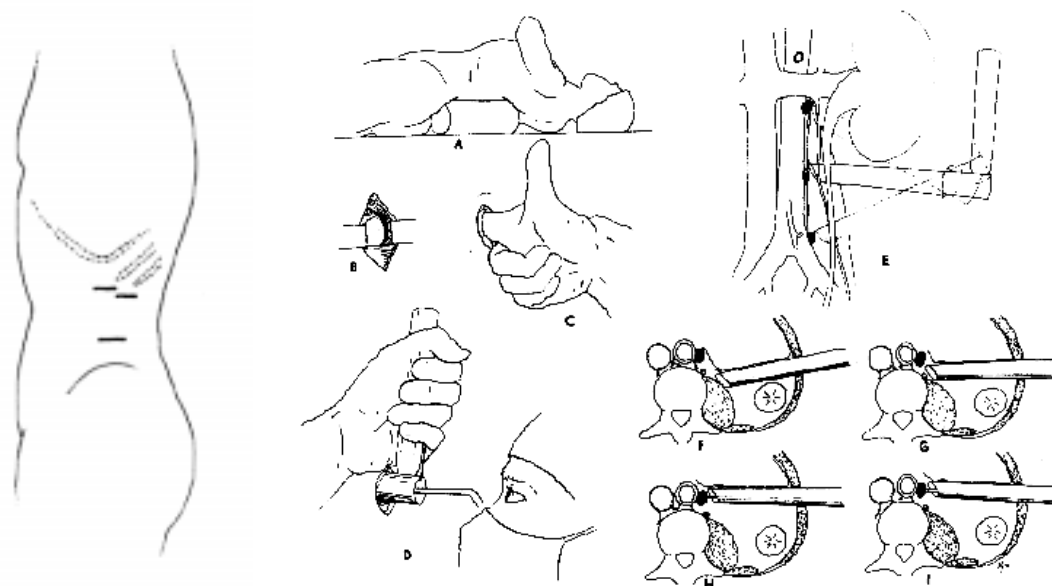


Fig. 4 - *Lomboscopie: voies d'abord et geste (d'après Sarrazin).*

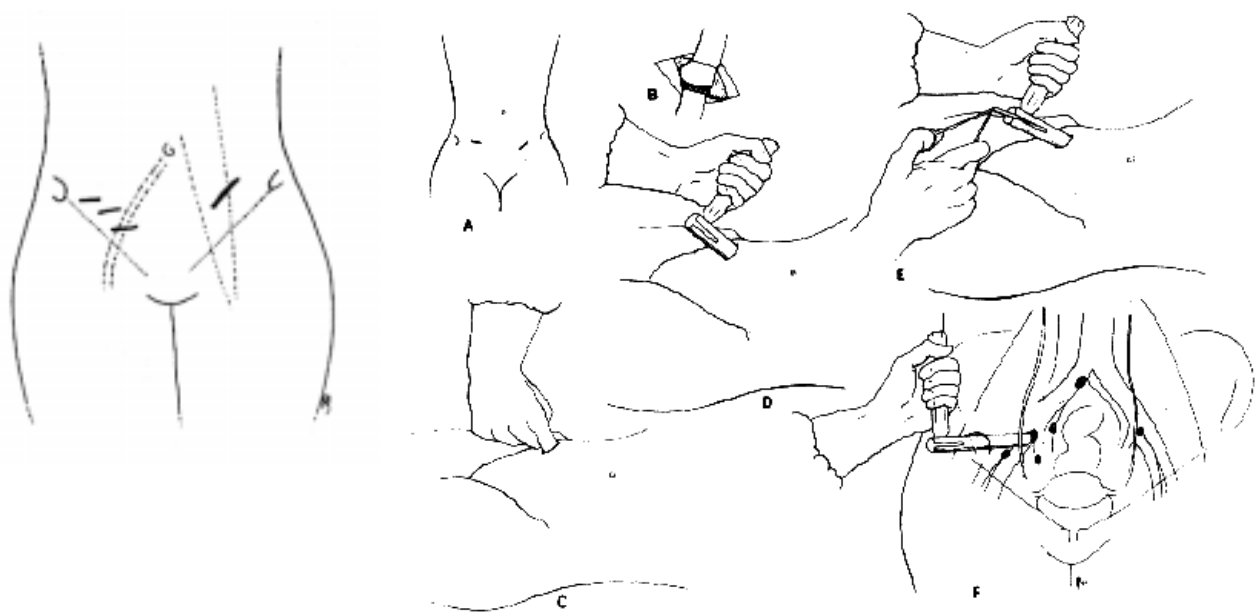


Fig. 5 - *Rétropéritonéoscopie iliaque: voies d'abord et geste (d'après Sarrazin).*

Le but de ces opérations est principalement diagnostique [96]. On les réalise fréquemment en chirurgie cancérologique au cours des bilans d'extension ou de récurrence des tumeurs solides et des hémopathies malignes (d'après [27]). Leur application

dans un cadre thérapeutique tend à se développer : lymphadénectomies pelviennes, drainage d'abcès profonds, sympathectomie lombaire.

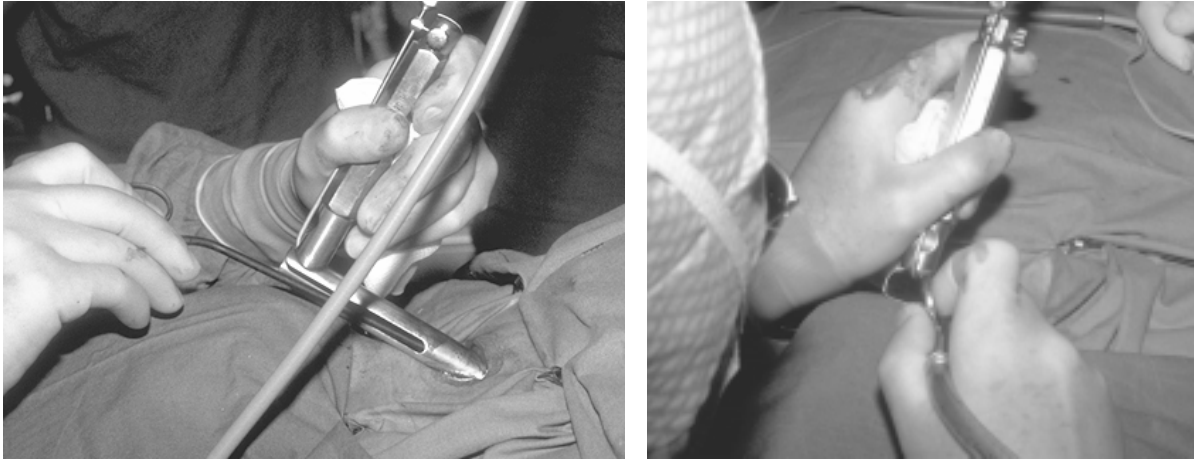


Fig. 6 - Médiastinoscopie (prof. Sarrazin).

Les instruments utilisés dans ces deux opérations sont identiques (figure 7). Parmi ceux-ci on trouve le médiastinoscope, tube rigide muni d'un manche et couplé à une source de lumière froide ; une échancrure permet l'utilisation de pinces à biopsie. Le médiastinoscope a été mis au point par Carlens en 1957.

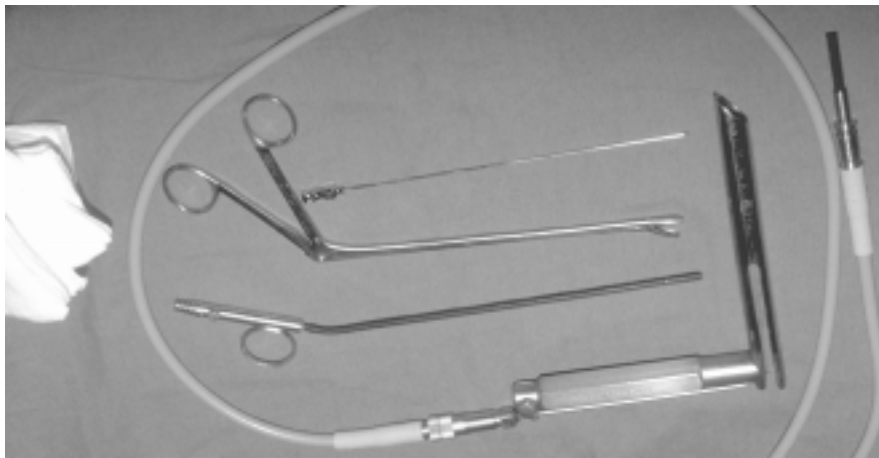


Fig. 7 - Instruments utilisés pour la rétroperitoneoscopie.

Le service de chirurgie générale et thoracique du CHU de Grenoble est un pionnier dans l'utilisation de ces techniques ; on y réalise annuellement une trentaine de rétroperitoneoscopies et une centaine de médiastinoscopies. Un système d'assistance rendant leur réalisation plus facile permettrait d'étendre leur utilisation.

La médiastinoscopie présente l'inconvénient de se situer dans une zone soumise aux mouvements respiratoires et aux déplacements dus aux battements cardiaques. Ces déplacements compliquent le calcul de la position de l'instrument par rapport

aux organes. C'est pour cette raison que notre premier système porte sur l'assistance de la rétroperitonéoscopie.

2.2 Matériel et système

Nous désirons fournir au chirurgien un suivi en temps réel des déplacements du médiastinoscope dans la région située au voisinage de la cible.

Pour ce faire nous avons réalisé aux cours de cette thèse différents prototypes de systèmes affichant en temps réel une série de coupes recalculées à partir d'un examen au scanner du patient (pour nos essais nous avons aussi utilisé l'imagerie par résonance magnétique nucléaire).

2.2.1 Repérage du médiastinoscope par rapport à l'image 3D

On connaît la géométrie du médiastinoscope dans un repère \mathcal{M} . Les coupes issues de l'examen du patient constituent un parallélépipède de voxels associé à un repère \mathcal{C} (figure 8). On souhaite mesurer la position de \mathcal{M} par rapport à \mathcal{C} . Pour ce faire on doit passer par des transformations intermédiaires dont le calcul nécessite 3 phases de calibrage.

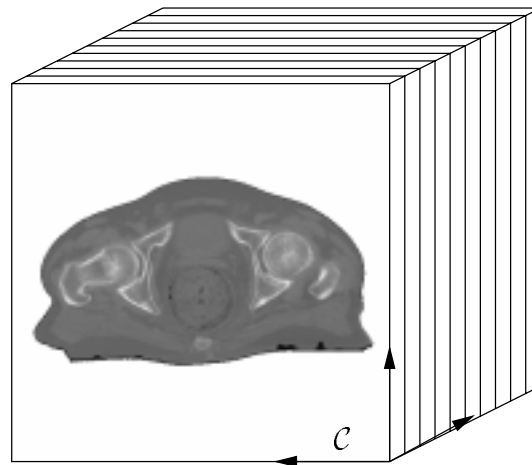


Fig. 8 - La position des organes du patient est connue dans une image tridimensionnelle (ici, des coupes scanner).

Les quatre repères intervenant dans cette section ainsi que les transformations calculées entre ces repères sont résumés sur la figure 9.

Localisateur 3D

La position du médiastinoscope est déterminée par un localisateur optique tridimensionnel : le système Optotrak(R). La partie optique de ce système est un banc rigide muni de trois capteurs comportant chacun une lentille hémicylindrique une barrette de capteurs CCD et un microprocesseur dédié au traitement de l'image

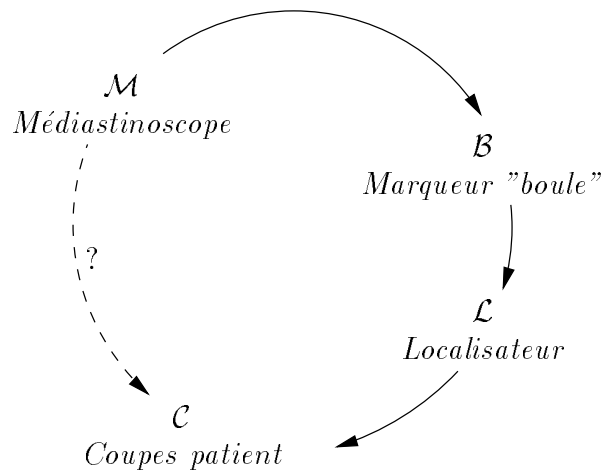


Fig. 9 - Repères et transformations intervenant dans le calcul de la transformation de \mathcal{M} vers \mathcal{C} , indiquée ici en pointillés.

unidimensionnelle fournie par la barrette. L'unité centrale du système active une par une des diodes émettant dans l'infra-rouge. Les points lumineux émis par ces diodes sont détectés par les capteurs. En utilisant les informations fournies par les trois capteurs le système calcule alors la position de chaque diode dans un repère lié au banc optique que nous noterons \mathcal{L} (figure 10).

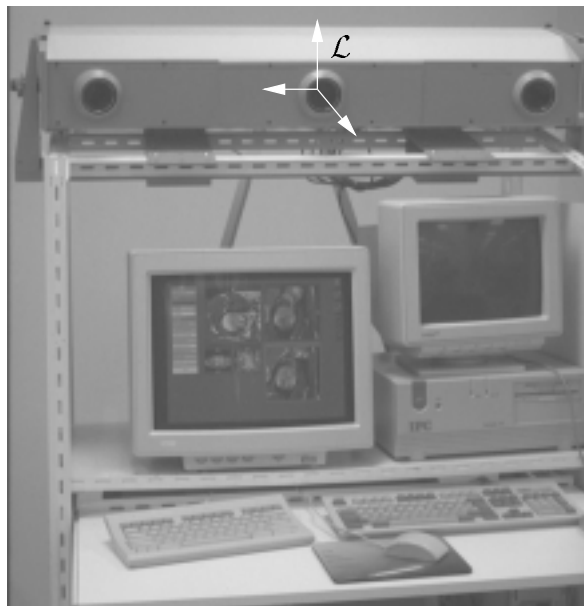


Fig. 10 - Le localisateur tridimensionnel Optotrak(R).

Marqueurs

Il suffit de fixer au moins trois diodes sur un solide rigide pour pouvoir ensuite déduire sa position à partir de celle des diodes.

Definition 1 (Marqueur)

Nous appellerons marqueur un solide rigide muni d'émetteurs ponctuels pouvant être localisés en 3D.

Afin de permettre le calcul de sa position Γ un marqueur doit être calibré. Ce calibrage consiste à associer un repère \mathcal{B} au marqueur et à établir les coordonnées b_i des émetteurs fixés sur le marqueur dans ce repère.

Une fois le calibrage effectué le localisateur mesure dans son repère \mathcal{L} les coordonnées l_i de ces mêmes points. La transformation permettant de passer de \mathcal{B} à \mathcal{L} est alors le meilleur déplacement tel que l'image de b_i soit proche de l_i pour tout i . Ce problème de mise en correspondance rigide de nuages de points est abordé dans le chapitre 4 de la partie *Outils*. Le système Optotrak(R) peut se charger de calculer ce déplacement Γ connaissant un calibrage du marqueur.

Si trois diodes suffisent pour effectuer ce calcul il est clair que plus le nombre de diodes visibles est élevé plus il devient précis. Il faut en outre s'assurer que le marqueur puisse être localisé dans toutes les positions qu'il sera amené à occuper. C'est pour ces raisons que nous avons conçu un marqueur en forme de boule muni de 24 diodes (figure 11). Un système de fixation spécial permet de lier rigidement cette boule à l'extrémité du manche du médiastinoscope. Nous notons \mathcal{B} le repère de calibrage du marqueur "boule".

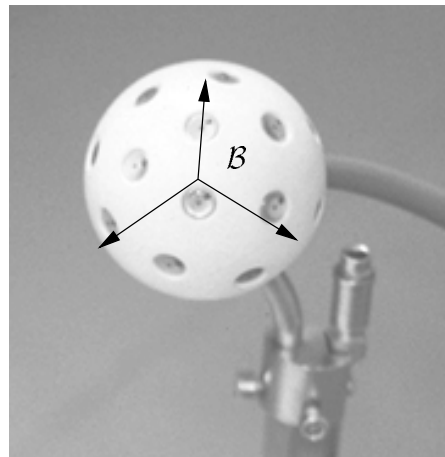


Fig. 11 - Le marqueur en forme de boule à 24 diodes et son repère de calibrage.

Socle de calibrage

Il est nécessaire de connaître la position du repère de calibrage de la boule \mathcal{B} par rapport au repère \mathcal{M} dans lequel la géométrie du médiastinoscope est donnée

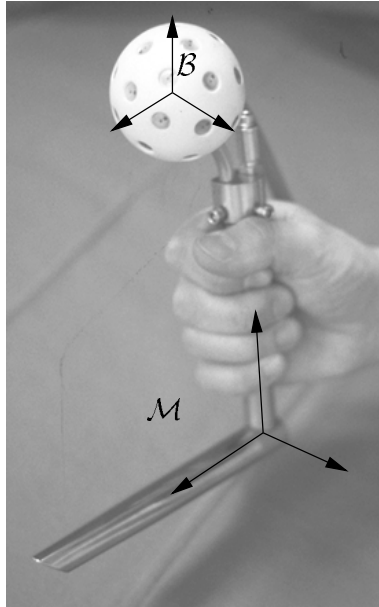


Fig. 12 - La géométrie du médiastinoscope est connue dans le repère \mathcal{M} .

(figure 12). C'est l'objet d'une seconde phase de calibrage dans laquelle on utilise un socle muni de marques géométriques connues par construction (rainures surfaces planes butées). Le socle est conçu pour que le médiastinoscope s'y emboîte de façon unique et reproductible et les marques permettent de calculer la position du socle par rapport au repère de la boule.

L'utilisation d'un tel socle a été requise par l'absence de surfaces caractéristiques sur le médiastinoscope à laquelle s'ajoute l'impossibilité de mettre le tube en contact avec des pointes risquant de rayer sa surface externe (qui doit glisser sur les organes lors de l'opération).

Recalage du patient

Il ne reste plus qu'à trouver la transformation entre le repère des coupes \mathcal{C} et le repère du localisateur \mathcal{L} . Le calcul de cette transformation est le but de la phase de recalage du patient. Pour effectuer ce recalage nous utilisons un système mis au point au laboratoire TIMC en collaboration avec le service de radiothérapie de l'hôpital de Grenoble par N. Laïeb, P. Vassal et J. Troccaz. Ce système met en correspondance la surface interne de l'os du bassin segmentée sur les coupes avec ces mêmes surfaces obtenues par échographie du patient pendant l'opération (voir partie *Outils* chapitre 4).

Les transformations de \mathcal{M} à \mathcal{B} et de \mathcal{L} à \mathcal{C} obtenues lors du calibrage alliées à la transformation de \mathcal{B} à \mathcal{L} mesurée en continu par le localisateur permettent de calculer en continu la transformation de \mathcal{M} à \mathcal{C} donnant la position des points du médiastinoscope dans le volume de coupes.

Nous utilisons cette transformation et le volume de données pour afficher en



Fig. 13 - *Le médiastinoscope sur son socle de calibrage.*

continu des informations supplémentaires sous forme d'images de synthèse.

2.2.2 Quelles images, sous quelle forme ?

Ce paragraphe étudie les différents types d'informations que l'on peut envisager de présenter au chirurgien ainsi que leur moyen d'affichage.

Plusieurs types de retour visuel sont possibles : d'une part sur la nature de l'immersion du chirurgien qui peut être complète (casque muni d'écrans ou *head mounted display*) partielle (écran semi-réfléchissant projection) ou nulle (affichage sur un écran externe) d'autre part sur la nature des images affichées : coupes recalculées de type scanner ou surfaces texturées.

Immersion complète

L'immersion complète n'est pas une solution viable à l'heure actuelle pour plusieurs raisons. Le casque est trop lourd pour être porté 6 heures par jour la résolution insuffisante des images entraîne une fatigue oculaire importante et l'immersion complète engendre des perturbations de l'équilibre liées à l'absence de coordination entre les sensations visuelles et les déplacements perçus par l'oreille interne. De plus le chirurgien ne peut pas accepter d'être privé de la vision directe : il a besoin de voir ses mains et les instruments qu'elles manipulent les instruments posés sur la tablette les examens affichés sur le mur les objets apportés par l'infirmière. L'immersion complète est cependant envisageable dans le cas de la télé-chirurgie.

Immersion partielle

Dans le cas d'une immersion partielle le champ visuel de l'opérateur (ici le chirurgien) est occupé par des images réelles auxquelles sont superposées des images de synthèse. Cette technique est utilisée depuis longtemps dans les avions de chasse

(viseur tête haute ou *head up display*). La superposition nécessite la connaissance simultanée de la position des yeux du chirurgien et de la transformation de recalage du patient vue plus haut. Un tel procédé utilisant un miroir semi-réfléchissant et un asservissement vidéo en continu a été développé par Peuchot et Tanguy [91].

Absence d'immersion

Il n'y a pas d'immersion dans le cas où les images de synthèse ne sont pas superposées à la scène réelle. C'est par exemple le cas dans les opérations sous contrôle vidéo.

Dans ce cas les images affichées peuvent être de différents types synthétisant plus ou moins les informations :

- Affichage tridimensionnel des surfaces des organes et des instruments [32-33].
- Affichage de coupes recalculées dans des plans évoluant avec les instruments il s'agit de l'approche utilisée dans nos expériences [5-6].
- Superposition d'images vidéo et de surfaces ou de coupes recalculées [58-108-19-115-61].
- Affichage de "viseurs" matérialisant la position courante de l'outil par rapport à sa position idéale. Ces viseurs sont affichés en deux ou trois dimensions. Les systèmes de guidage passif développés par Sautot [97] pour le vissage pédiculaire, Orti et Dessenne [89-36] pour le remplacement du ligament croisé antérieur du genou et le remplacement du condyle en chirurgie maxillo-faciale utilisent ce type d'affichage.

2.2.3 Système d'assistance proposé

Choix du type d'images

Dans les différents prototypes développés nous utilisons un affichage sans immersion. Les images affichées en continu sont des coupes de type scanner recalculées dans des plans a priori quelconques à partir de l'examen scanner du patient. Les plans de coupe se déplacent en fonction de la position de l'instrument mesurée en continu par le localisateur.

L'utilisation de coupes planes de type scanner apporte des informations plus lisibles pour le chirurgien que l'affichage de surfaces colorées des différents organes.

Après quelques essais il s'avère que l'interprétation des images par le chirurgien est bien plus facile quand les coupes sont parallèles aux directions "classiques" (axiales, frontales et sagittales). Outre ces coupes classiques nous avons tout de même retenu l'affichage d'un plan de coupe perpendiculaire à l'axe du tube et passant par son extrémité interne. A part celui-ci nous présentons pour chaque direction de coupe une vue globale et une vue locale grossie centrée sur l'extrémité interne de l'outil. Sur chaque image est dessinée une projection de l'axe du tube.

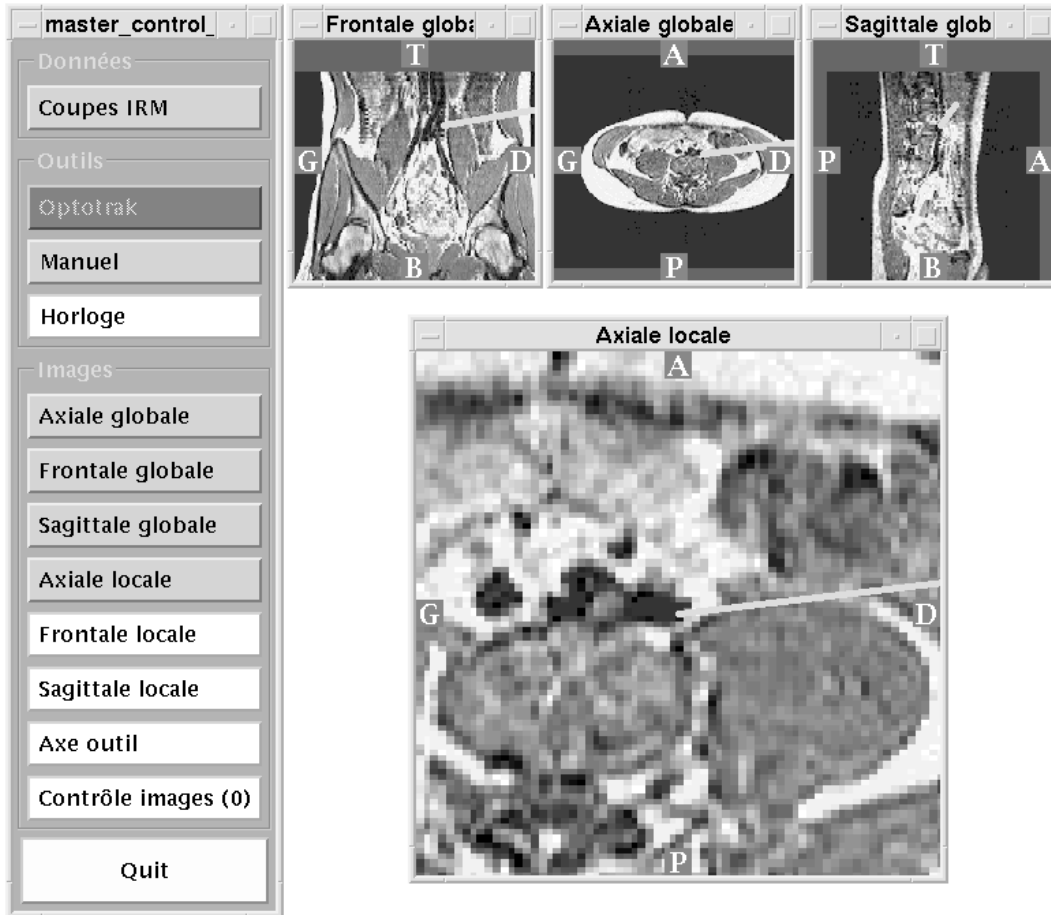


Fig. 14 - Images affichées en cours d'opération. Le segment clair symbolise l'axe de l'outil.

Architecture logicielle

Le choix du type d'images affichées rend nécessaire le calcul en continu de plusieurs coupes dans des plans a priori quelconques. Nous avons choisi de laisser le système répartir le temps du processeur entre les différentes images : chaque image est associée à un processus différent et tous ces processus tournent en même temps.

L'examen scanner du patient prenant de l'ordre de 10 Mo il n'est pas possible que chaque processus affichant une image le charge dans son espace mémoire : l'examen scanner est donc chargé dans une zone de mémoire partagée et utilisée simultanément par tous les processus affichant une image. Un processus particulier est chargé d'initialiser cette zone de mémoire.

La communication de la position du médiastinoscope aux processus affichant les images s'effectue elle aussi par partage d'une zone de mémoire. Un processus particulier est chargé de communiquer avec le localisateur pour mesurer la position de \mathcal{B} dans \mathcal{L} et en déduire la position de \mathcal{M} dans \mathcal{C} en utilisant les résultats des différents calibrages. Ce procédé rend très simple l'enregistrement et la restitution de

séquences de positions : il suffit de lancer un processus indépendant qui transfère en continu les valeurs de la position entre la zone de mémoire et le disque.

Le lancement et l'arrêt de ces processus sont gérés par un processus "pilote". La communication entre les processus s'effectue par des messages (mécanisme fourni par le système).

Le premier processus lancé par le pilote affiche une interface permettant de commander le lancement et l'interruption des autres types de processus. Cette interface apparaît dans la fenêtre de gauche de la figure 14.

La figure 15 résume les différents processus (boîtes arrondies) et données entrant en jeu dans le système. Le canal de communication gris représente la communication entre le pilote et les autres processus ; les flèches symbolisent les échanges de données.

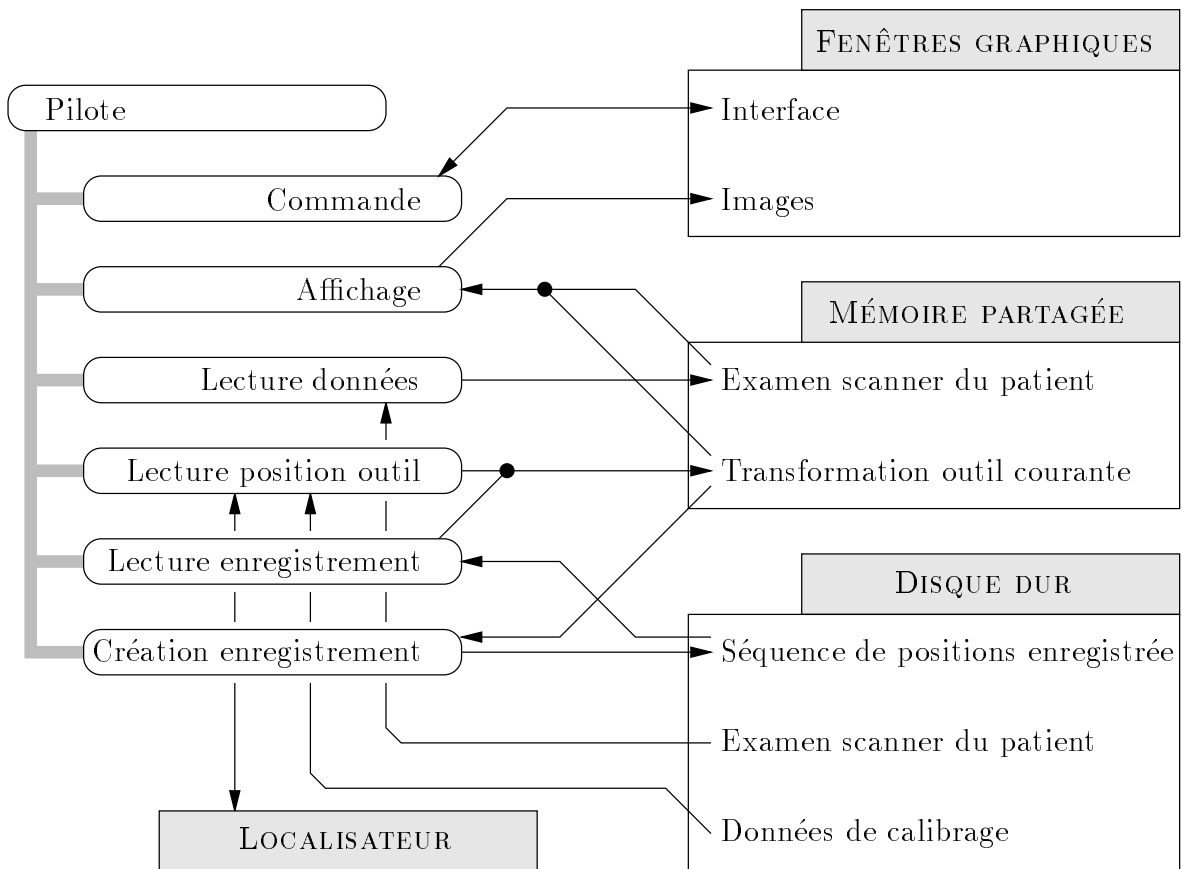


Fig. 15 - Les différents types de processus et de données du système.

2.3 Protocole

Nous ne décrivons ici que les phases de l'opération liées à l'apport de l'assistance informatique. L'opération se décompose en trois phases. La phase pré-opératoire a lieu en dehors du bloc opératoire. La phase de recalage du patient et de calibrage

des instruments a lieu au bloc entre l'anesthésie du patient et le geste chirurgical. La phase d'assistance per-opératoire se déroule pendant le geste.

2.3.1 Phase pré-opératoire

La phase pré-opératoire comprend la stérilisation des instruments, l'acquisition d'images de la région anatomique concernée et leur pré-traitement.

Le médiastinoscope et sa boule de repérage doivent être démontés et stérilisés chacun selon deux procédés différents ainsi que son socle de calibrage et les petits outils nécessaires à l'assemblage et au calibrage.

Une série de coupes de la zone à opérer est acquise au scanner. Cet examen se pratique systématiquement avant toute rétro-péritonéoscopie classique. Les images sont ensuite transférées sur le disque dur de la station de travail.

A partir de ces images on construit une représentation par un nuage de points de la surface interne de l'os du bassin. D'autre part le chirurgien identifie les cibles de l'opération. Ces deux informations sont alors connues dans le repère lié aux images \mathcal{C} .

2.3.2 Calibrage du médiastinoscope

Le calibrage du médiastinoscope au début de l'opération est rendu nécessaire par son démontage pendant la stérilisation.

Ce calibrage nécessite l'utilisation de deux marqueurs supplémentaires qui doivent eux aussi être stérilisés. Le premier est fixé sur le socle de calibrage et sert de référence (ceci permet de déplacer le socle par rapport au localisateur pendant le calibrage). Le second est muni d'une pointe et sert à désigner des points du socle.

Les trois marqueurs (la boule et les deux marqueurs auxiliaires) sont tout d'abord calibrés.

Le premier marqueur auxiliaire est fixé sur le socle. La pointe du second marqueur est calibrée : on obtient les coordonnées de la pointe dans le repère de calibrage du marqueur.

La pointe est alors utilisée pour obtenir les coordonnées du plan du socle et des trois rainures gravées sur ce plan. Ceci permet de connaître la position du repère dans lequel la géométrie du socle est donnée. Ensuite on insère le médiastinoscope dans le socle dans l'unique position possible. On peut alors déduire la transformation de \mathcal{M} à \mathcal{B} .

2.3.3 Recalage du patient

Le recalage du patient est effectué selon la procédure développée par N. Laïeb, P. Vassal et J. Troccaz pour la radiothérapie conformationnelle de la prostate [107].

Le recalage est effectué juste après l'anesthésie du patient qui est alors immobilisé sur la table d'opération.

On utilise un échographe muni d'un marqueur (figure 16) pour acquérir des images de la surface antérieure de l'os du bassin au voisinage des crêtes iliaques. L'utilisation du marqueur permet après un calibrage préalable (réalisé avant l'opération)

d'obtenir la position tridimensionnelle dans le repère du localisateur \mathcal{L} d'un point quelconque de l'image échographique.

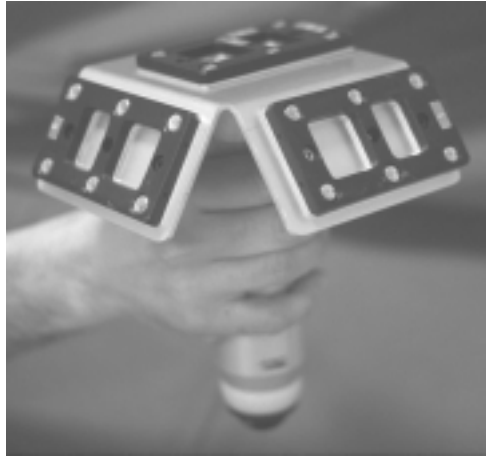


Fig. 16 - *Echographe muni de marqueurs utilisé pour numériser la surface de l'os du bassin.*

La segmentation des images échographiques permet ainsi d'obtenir des courbes dans \mathcal{L} . Ces courbes sont mises en correspondance avec la surface de l'os segmentée dans \mathcal{C} lors du traitement pré-opératoire. Nous employons un algorithme de mise en correspondance décrit par Lavallée et Szeliski dans [74]. On calcule ainsi la transformation de \mathcal{L} dans \mathcal{C} .

Cette transformation est valide tant que le patient n'est pas déplacé par rapport au localisateur. Cette contrainte peut être évitée en fixant un marqueur sur le bassin pour servir de repère de référence à la place de \mathcal{L} .

2.3.4 Navigation assistée

Prenant en entrée les images du patient et les différentes transformations vues précédemment le système de navigation assistée ne requiert aucune manipulation. L'interface de la dernière version permet de choisir les vues à afficher d'enregistrer et de rejouer des séquences de mesures.

2.4 Expérimentation

Nous avons réalisé quatre versions majeures du logiciel d'assistance.

La première écrite en novembre 1992 a servi à démontrer la faisabilité de notre approche.

Lors de l'expérience réalisée avec cette première version en décembre 1992 le "patient" était un mannequin en plastique démontable (utilisé pour illustrer les cours d'anatomie) dans lequel nous avons collé des billes métalliques de 2 mm de diamètre. Le mannequin était ensuite passé au scanner. L'expérience consistait à diriger vers les cibles la pointe d'un marqueur cylindrique introduit dans le mannequin par une petite ouverture. Dans cette version le recalage du patient était effectué à

l'aide de billes métalliques insérées dans le mannequin avant le scanner et désignées lors de l'opération avec une pointe munie d'un marqueur ; il suffit de mettre en correspondance les positions des billes pour obtenir la transformation de recalage. L'algorithme de détection des billes métalliques est présenté dans le chapitre 5 de la partie *Outils*.

La seconde version écrite au début de 1993 a permis de tester en conditions réelles la précision et l'utilité de l'assistance.

Fin août 1993 nous avons testé cette version du système sur une pièce anatomique grâce à la collaboration du service de radiologie centrale de l'hôpital de Grenoble et du laboratoire d'anatomie de la faculté de médecine de Grenoble. Cette expérience a permis de vérifier la précision du repérage des structures osseuses et de constater les déplacements (dus à la présence du médiastinoscope) de la poche péritonéale. La position des cibles insérées au voisinage des ganglions habituellement visés par la rétopéritonéoscopie iliaque correspondait aux informations fournies par le système. Le recalage du patient était effectué par la même méthode que dans la première version (insertion de plombs sous la peau au niveau des crêtes iliaques et de la symphyse pubienne).

La troisième version datant de 1994 incluait les modifications inspirées par l'expérience sur cadavre : utiliser la méthode de recalage du patient par échographie développée pour la radiothérapie améliorer la vitesse de l'affichage concevoir le marqueur en forme de boule et remplacer le socle de calibrage en bois par un modèle stérilisable. De plus cette version était la première mouture de l'architecture logicielle présentée plus haut. Nous présentons ce système en détails dans [626].

La quatrième version écrite au début de 1995 est une mise au propre de la version précédente : utilisation modulaire et plus efficace du système d'exploitation et de l'interface graphique fichiers de configuration... La modularité du système a permis son utilisation sur différents types d'examen et d'outils en particulier en ORL.

2.5 Conclusion

A travers l'exemple de la rétopéritonéoscopie nous avons vu les problèmes liés à la conception la réalisation et l'expérimentation de systèmes d'assistance à une opération chirurgicale.

La conception d'un tel système passe par la prise en compte des points suivants :

- la nature du geste et de ses difficultés les souhaits des chirurgiens
- les données que peut apporter l'assistance pour faciliter ou rendre plus précise l'opération
- les instruments permettant de mesurer ces données (capteurs spéciaux calibrage...) et de les restituer (retour visuel mais aussi sonore et tactile).

La réalisation du système nécessite alors :

- une architecture logicielle adaptée
- la fabrication des instruments de mesure (par exemple le marqueur en forme de boule et le socle de calibrage)
- la mise au point des procédures de calibrage.

L'expérimentation implique :

- la modification du protocole opératoire
- la collaboration des différents services de l'hôpital et des personnels du bloc opératoire.

En ce qui concerne le système que nous avons développé, les expériences ont prouvé la faisabilité et l'utilité d'une telle assistance. La modularité du système a permis son utilisation dans d'autres contextes.

Chapitre 3

Conclusion

Dans cette partie nous avons décrit nos travaux concernant la conception, la réalisation et l'expérimentation d'un système d'assistance passive d'une opération chirurgicale : la rétro-péritonéoscopie.

Ce système intègre une architecture logicielle modulaire qui le rend utilisable directement dans d'autres types d'opérations. Nous avons conçu un prototype d'outil chirurgical "augmenté" et un socle permettant de le calibrer.

Des expériences en conditions réelles ont prouvé l'utilité d'un tel système.

Cette étude a permis d'identifier plusieurs problèmes que l'on retrouve dans la plupart des applications de chirurgie assistée par ordinateur :

- Le calibrage des capteurs. Il apparaît ici sous deux formes : le calibrage des marqueurs et la mesure précise de structures géométriques (plans, rainures, points).
- La mise en correspondance utilisée pour le recalage du patient.
- La détection d'objets sphériques dans une image volumique (ou d'objets circulaires dans une image plane) utilisée dans la localisation précise des centres des billes métalliques lors de nos premières expériences.

Ces deux derniers problèmes seront abordés dans la partie *Outils*.

D'autre part, la simulation du geste passe par l'utilisation d'un modèle physique des organes avec lequel l'opérateur peut interagir en manipulant des instruments "virtuels" incluant des sensations de résistance à l'effort. Ceci a motivé une étude des différents modèles physiques du point de vue de l'animation et du modelage assistés par ordinateur dans la partie *Animation de systèmes de solides* et aussi du point de vue théorique dans le chapitre 3 de la partie *Outils*.

Deuxième partie

Animation de systèmes de solides

Chapitre 1

Introduction

Dans cette partie nous nous intéressons à la simulation de systèmes de solides rigides ou déformables pour l’animation ou le modelage (*modelling*). Bien que cette étude soit générale elle est effectuée en gardant en tête notre objectif final : la réalisation d’un simulateur de geste chirurgical.

Pour simuler le comportement d’un tel système il faut tout d’abord le modéliser c’est-à-dire :

- représenter sa **géométrie** et ses **déformations** (au sens large ce qui inclut aussi les déplacements)
- fixer les **contraintes** que doivent vérifier ces déformations (par exemple imposer la non-interpénétration des différents solides)
- donner les règles du **contrôle** de l’évolution du modèle (dans le cas de la contrainte de non-interpénétration le contrôle doit définir la façon dont le système réagit lors de la collision entre deux solides).

Une fois le système modélisé il faut simuler numériquement son évolution en suivant les règles de contrôle et en respectant les contraintes.

Cette partie débute par une étude bibliographique des systèmes existants d’animation et de modelage par ordinateur contrôlés en partie ou en totalité par les lois de la physique (chapitre 2). Dans cette présentation nous avons tenté d’unifier les problèmes voisins de l’animation et du modelage de forme et d’identifier pour chaque publication les trois composantes vues plus haut : géométrie et déformations contraintes contrôle.

Dans le chapitre suivant (chapitre 3) nous définissons les fonctionnalités que doit posséder un modèle “géométrique et dynamique” de simulation de geste chirurgical. Géométrie parce que le modèle doit permettre le calcul et l’évolution de la géométrie des différents organes intervenant dans l’opération. Dynamique parce que cette évolution doit se faire en respectant les lois de la physique en particulier en ce qui concerne le comportement déformable des organes et la gestion des contacts.

Après avoir défini ces fonctionnalités nous ferons un compte-rendu des travaux et expériences que nous avons effectués en vue de réaliser un tel modèle en partant de l’hypothèse d’une évolution quasi-statique du système. Nous exposons comment on

peut séparer effectivement les trois composantes dégagées plus haut : géométrie et déformations, contraintes et contrôle. Cette approche modulaire permet d'utiliser simultanément plusieurs représentations des solides dans le même système. Nous utiliserons des lois de comportement non linéaires et des formulations permettant de grands déplacements des solides, ce qui n'a du moins à notre connaissance pas encore été le cas pour l'animation de solides par ordinateur.

Dans la conclusion de cette partie (chapitre 4) nous ferons la synthèse de ces expériences et nous verrons les problèmes restant à explorer dans ce domaine.

Chapitre 2

Animation et modelage par ordinateur

Dans ce chapitre nous exposons les différents choix adoptés dans la littérature concernant la conception et la réalisation de systèmes d'animation ou de modelage.

La réalisation de tels systèmes passe par la définition de trois composantes : la représentation de la géométrie et des déformations, les contraintes s'appliquant au système et la stratégie de contrôle du modèle.

Ces termes reprenant les notions introduites par Wilhelms dans [111] sont définis dans la première section. Pour chacune de ces trois composantes nous détaillons ensuite les différents choix présentés dans la littérature en examinant plus particulièrement les modèles de solides déformables.

2.1 Définitions

La déformation associe tout point du solide dans une configuration de référence à sa position courante dans l'espace (voir le chapitre 3 de la partie *Outils*). Cette définition inclut aussi bien les changements de forme des solides que leurs changements de position ; c'est pourquoi nous emploierons aussi ce terme dans le cas de solides rigides (l'application est alors une isométrie directe).

Definition 2 (Déformation)

Une déformation est une application suffisamment régulière pour que les grandeurs que nous en dérivons soient correctement définies, injective et préservant l'orientation de l'espace (voir [30]).

Definition 3 (Géométrie)

Nous appellerons géométrie la surface extérieure du solide pour une déformation donnée. Nous étendrons ce terme à la structure de données permettant d'associer la géométrie à la déformation.

Une représentation discrète – c'est à dire par un nombre fini de paramètres réels – de la géométrie des solides et de leurs déformations est nécessaire à leur traitement par ordinateur.

A chaque solide est associé une déformation. Elle-même paramétrée par un nombre fini de variables. Nous pouvons ainsi définir l'état du système :

Definition 4 (Etat)

L'état du système est un vecteur rassemblant les paramètres des déformations de tous les solides le constituant. Pour les systèmes évoluant selon des équations différentielles du second ordre on ajoutera à l'état les dérivées d'ordre 1 de ces paramètres par rapport au temps (vitesses et vitesses angulaires).

L'opérateur (l'animateur ou le modelleur utilisant le système) restreint l'ensemble des états que peut occuper le système et doit définir l'algorithme dirigeant son évolution. On retrouve ces notions en robotique.

Definition 5 (Contrainte)

Une contrainte est une équation ou inéquation portant sur l'état du système et éventuellement sur ses dérivées spatiales ou temporelles. Nous élargirons cette définition à des prédicats quelconques par exemple la condition de non-intersection des intérieurs des objets.

Definition 6 (Espace libre)

L'espace libre est l'ensemble des états du système vérifiant les contraintes.

Definition 7 (Contrôle)

Le contrôle est l'algorithme fixant l'évolution de l'état du système en tenant compte des contraintes.

Dans le cas de l'animation le contrôle affecte l'évolution du système au cours du temps. Dans le cas du modelage de formes le contrôle définit la stratégie d'optimisation permettant d'obtenir la forme recherchée : celle qui vérifie au mieux les contraintes.

2.2 Déformations et géométrie

La représentation des déformations est liée à la nature des objets modélisés. Nous distinguerons le cas des solides ponctuels formant les systèmes de particules et des solides rigides articulés et des solides déformables.

2.2.1 Solides rigides articulés

A la fin des années 1970 et au début des années 1980 la simulation du comportement des robots et l'animation de personnages et animaux ont motivé l'intérêt pour les systèmes de solides rigides reliés par des articulations. Featherstone présente en 1983 dans [48] un algorithme de simulation numérique du comportement d'un tel système ; on y trouve les références des travaux antérieurs non cités ici.

Déformations

Décrire la position d'un seul solide rigide nécessite 6 paramètres en dimension 3 et 3 paramètres en dimension 2. Chaque liaison entre deux solides retire un certain nombre (1 à 6) de degrés de liberté au système. Usuellement on considère des articulations possédant un seul degré de liberté soit en rotation soit en translation ce qui permet de modéliser la grande majorité des articulations animales et robotiques. De plus une liaison à $d \geq 2$ degrés de liberté peut être obtenue en juxtaposant d liaisons à un degré de liberté reliant des solides intermédiaires “virtuels”.

La position du système est donc représentée par celle d'un des solides du système et les coordonnées articulaires de chacune des liaisons. Schroeder et Zeltzer [98] utilisent une représentation homogène à 6 degrés de liberté pour chaque solide chaque liaison ajoutant une nouvelle contrainte au système.

Géométrie

Les surfaces des solides composant le système n'introduisent des degrés de liberté dans le système que dans le cas des modèles hybrides où les solides ne sont pas rigides. Chadwick Haumann et Parent [25] fixent des muscles sur un squelette articulé. Les muscles sont représentés par des grilles tridimensionnelles de points de contrôle définissant une déformation de type *Free Form Deformation* (FFD) [99]. Certains points de contrôle sont fixés au squelette et la déformation est soumise à des lois de comportement modélisant l'action des muscles. La peau est dessinée en suivant les mêmes déformations. Gascuel dans [50 52 51] entoure des squelettes articulés de masses reliées au squelette par des ressorts. La peau est définie comme une surface d'isovaleur d'une fonction définie par la position des masses. Ce modèle permet d'obtenir des contacts étendus entre objets. Ertl *et al.* [45] rajoutent des masses mobiles autour des segments d'un mannequin articulé. L'adjonction de ces masses augmente peu le nombre de degrés de liberté au système puisqu'on ne considère que la position de leur centre de gravité par rapport aux segments sur lesquels elles sont fixées.

2.2.2 Systèmes de particules

En 1983 Reeves décrit dans [94] un système de particules qu'il utilise pour simuler la propagation d'un mur de feu sur une planète dans le film *Star Trek II*.

Déformations

La position d'une particule est représentée par un point. Une masse est associée à chaque particule. Szeliski et Tonnesen [101] ajoutent à ces paramètres un vecteur unitaire définissant une orientation intérieur/extérieur pour chaque particule.

Bien que les particules puissent être utilisées individuellement par exemple pour visualiser des écoulements résultant de simulations elles sont fréquemment reliées par des interactions de type ressort/amortisseur pour créer des “solides particuliers”.

Miller [82] anime de façon très réaliste des serpents par une “ligne” de masses ponctuelles reliées par des ressorts. Dumont, Arnaldi et Hégron [42] modélisent des solides déformables par des masses reliées par un réseau cubique tridimensionnel de ressorts. House et Breen [66] proposent l’utilisation de systèmes de particules pour modéliser le comportement de tissus mous lors d’un geste chirurgical. Dans [88], Norton *et al.* proposent d’utiliser un système de particules pour permettre la fracture des objets lors de collisions. Delingette *et al.* [33] manipulent des surfaces à facettes polygonales dont le dual est une triangulation ou leur analogue tridimensionnel ; la surface constitue un système de type masses/ressorts. Lombardo [79] étend le modèle de Szeliski et Tonnesen en y ajoutant des contraintes permettant à la surface de retrouver une forme de référence.

Géométrie

La peau des serpents de Miller est constituée de carreaux de surfaces splines dont les points de contrôle sont des masses ponctuelles. Jokhadar [68] dessine une sphère autour de chaque particule ; la surface des objets est alors celle de la réunion des sphères. Van Overveld [110] construit un champ de déplacement à partir des positions des particules ; ce champ est ensuite utilisé pour déplacer un polygone définissant la frontière des objets.

Le nombre des particules est en général fixe, sauf dans le modèle de Szeliski et Tonnesen dans lequel on cherche à conserver un maillage régulier des surfaces.

2.2.3 Surfaces et volumes déformables

En 1987, Terzopoulos [104] présente un modèle continu de solides déformables. Dans ce modèle, des objets parallélépipédiques sont représentés par un maillage d’éléments cubiques. A chaque déformation est associée une énergie potentielle qui définit l’évolution du système.

Différents modèles de solides déformables “continus” ont ensuite été présentés. Ces modèles sont utilisés non seulement pour simuler l’évolution de systèmes de solides déformables, mais aussi pour modéliser la matière et créer de nouvelles formes en tenant compte de contraintes géométriques.

Le modèle continu a une infinité de degrés de liberté : l’état du système est une fonction continue associant à chaque point du système dans une configuration de référence sa position dans la configuration courante.

Dans certains modèles, seule la surface extérieure du solide est prise en compte et déformée. Dans les autres, la déformation est définie en tout point de l’intérieur du solide.

Dans les deux cas, il faut discrétiser la déformation. Quatre approches ont été proposées : les déformations globales de l’espace entourant le solide, les déformations définies par morceaux dans l’intérieur ou sur la surface du solide, les solides ou surfaces paramétrés, les surfaces implicites.

Déformations globales

Les déformations globales sont définies dans une portion d'espace englobant la configuration de référence. Elles permettent de découpler la déformation et la géométrie du solide. En effet on peut imaginer l'objet pris dans un bloc de gélatine déformable : on déforme la gélatine et l'objet suit les déplacements. La géométrie de l'objet déformé est donc l'image de la géométrie initiale par la fonction de déformation du bloc de gélatine.

Les déformations globales peuvent être définies de différentes façons :

- Par un treillis de contrôle de type FFD (*free form deformation*) [99] ou EFFF (*extended free form deformation*) [31] dans lequel on déplace l'ensemble des points contrôlant une fonction d'interpolation tridimensionnelle de type spline [113].
- Par ses coefficients dans une combinaison linéaire de fonctions de base [21]. Ces fonctions peuvent être par exemple des polynômes en (x, y, z) de faible degré. Un choix judicieux présenté dans [90] est de prendre les déformations correspondant aux modes propres de la fonctionnelle d'énergie potentielle élastique *du bloc de gélatine* en ignorant les modes de hautes fréquences ce qui permet de réduire la dimension du vecteur d'état et ainsi d'accélérer la simulation.

Déformations définies par morceaux

Le solide est découpé en volumes élémentaires (les éléments) comme les cubes ou les tétraèdres. Les paramètres de la déformation sont les positions des sommets des éléments et d'éventuels autres points situés aux centres des facettes par exemple.

La déformation est alors définie sur chaque élément par des fonctions d'interpolation polynomiales. La continuité entre les éléments est garantie par construction [56].

Dans le cas d'un modèle surfacique c'est la frontière du solide qui est découpée en éléments (carrés, triangles...) [24].

Cette représentation des déformations sert de base à la résolution par la méthode des éléments finis des problèmes dont l'inconnue est un champ défini dans l'objet ou sur sa surface (déplacements et contraintes pour le calcul de structures, champs électro-magnétiques dans les conducteurs, champs de vitesse pour le calcul d'écoulements).

Surfaces paramétrées

La surface du solide peut être définie par une fonction paramétrée comme dans le cas des superquadriques. La déformation est alors obtenue en modifiant les valeurs des paramètres.

Surfaces implicites

La surface du solide est dans ce cas définie par une équation du type $f_p(M) = 0$ [50Γ52Γ51]. p est un vecteur de paramètres représentant par exemple la déformation d'un squelette rigide articulé autour duquel est défini le potentiel f_p .

Déformations hybrides

Terzopoulos et Fleischer [103] représentent la déformation sur deux niveaux : tout d'abord un déplacement rigideΓet ensuite un changement de forme autour de cette position. Ceci permet de conserver une définition simple (linéaire) de l'énergie potentielle élastique ne dépendant que du second termeΓtout en permettant de grands déplacements *via* le terme correspondant au déplacement rigide.

2.3 Contraintes

Nous employons le terme *contrainte* dans le sens de la définition 5 donnée au début de ce chapitre. En modelageΓles contraintes permettent de traduire les idées du concepteur en équationsΓqui seront ensuite prises en compte par l'algorithme de contrôle pour fournir une forme.

En animationΓle script (trajectoiresΓinteractions) prévu par l'animateur est traduit en contraintes. D'autres contraintes portant sur le comportement des solides permettent d'obtenir des mouvements plus réalistes.

Nous identifions dans cette section les attributs des contraintes rencontrées dans la littérature. EnsuiteΓnous exposons les différents types de contraintes existantsΓclassés en fonction de ces attributs.

2.3.1 Ordre : géométrique/cinématique/dynamique

Selon qu'une contrainte porte sur les paramètres de la déformation (positions)Γsur ses dérivées d'ordre 1 (vitesses) ou sur ses dérivées d'ordre 2 (accélérations)Γon la qualifie de géométriqueΓcinématique ou dynamique.

2.3.2 Portée : fixe/conditionnelle

Une contrainte fixe consiste en une ou plusieurs équations que le système doit vérifier à chaque instant.

Une contrainte conditionnelle n'impose aucune équation jusqu'à ce qu'une condition (la condition d'établissement) ne provoque son apparition. A partir de cet instantΓde nouvelles équations doivent être vérifiées par le systèmeΓjusqu'à ce qu'une autre condition (la condition de rupture) annule la contrainte.

2.3.3 Formulation : forte/faible

Nous distinguons deux formulations des contraintes. La formulation forte impose la vérification des équations de la contrainte par l'état du système. La formulation faible introduit ces équations à travers un terme "énergétique" minimal quand

elles sont vérifiées. Ces “énergies” ne sont pas toujours effectivement des grandeurs physiques homogènes à des énergies. On retrouve ces notions dans les méthodes d’éléments finis [116Γ117Γ15] et la théorie de l’élasticité (voir le chapitre 3 de la partie *Outils*).

Les contraintes fortes réduisent la dimension de l’espace libreΓce qui n’est pas le cas pour les contraintes faibles. C’est pour cette raison qu’en présence d’un grand nombre de contraintesΓseule la formulation faible est possibleΓl’ensemble des contraintes ne pouvant être simultanément vérifié.

Les contraintes faibles permettent d’associer une “énergie” à chaque état ; l’espace libre correspond aux cuvettes de potentiel de cette énergie.

Les contraintes fortes fixes déterminent la dimension de l’espace libreΓ et les contraintes fortes conditionnelles en définissent les bords.

2.3.4 Interactivité

Certains paramètres de l’état du système peuvent être liés à des grandeurs extérieures. Nous parlerons dans ce cas de contraintes interactives. C’est en particulier le cas quand la contrainte est fournie par un système de mesure de position ou d’effort manipulé par l’opérateur.

2.3.5 Globalité

Dans certains casΓil peut s’avérer intéressant de rechercher une trajectoire optimisant une contrainte globale du type plus court chemin en distance ou en tempsΓ plus faible travail musculaire.

Ce type de contrainte permet d’obtenir des trajectoires très réalistes pour les comportements du vivant comme la marche [23]Γla nageΓla reptation [82]. Witkin et Kass optimisent les sauts d’une “lampe de bureau” (Luxo) à quatre articulations dans [112].

2.3.6 Exemples de contraintes

Contraintes géométriques entre particules

Dans le cas de systèmes de particulesΓon peut imposer des contraintes du type “les particules A et B sont à une distance d_0 donnée”. Il s’agit d’une contrainte géométrique fixe. Sa formulation forte correspond à l’équation $\|A - B\| = d_0$ Γet sa formulation faible à un terme du type $(\|A - B\| - d_0)^2$ à minimiser.

Cette contrainte et d’autres du même type sont exploitées par Luciani [80]ΓSzeliski et Tonnesen [101]ΓLombardo [79] pour construire des objets complexes en agrégeant des particules.

Liaisons entre solides

Dans un système de solides articulésΓles liaisons entre les solides constituent des contraintes géométriques fixes.

Contacts fixes

Dans tous les modèles pour lesquels on peut définir la notion de surface des solides Γ on peut imposer des contraintes géométriques fixes de contact entre des parties (points Γ courbes Γ portions de surface) de deux objets Γ par exemple :

- le point A de l'objet 1 est confondu avec le point B de l'objet 2 Γ
- le point A de l'objet 1 est sur la surface S définie sur l'objet 2.

Borrel et Rappoport [21] Γ Gascuel et Gascuel [49] utilisent ce type de contraintes.

Inégalités portant sur les paramètres géométriques

Les inégalités se traduisent par des contraintes conditionnelles. Par exemple si on impose à l'angle θ d'une articulation de vérifier $\theta_1 \leq \theta$ l'inégalité peut être vérifiée et n'entraîner aucune contrainte Γ jusqu'à ce que la condition d'établissement $\theta_1 \geq \theta$ soit vérifiée. Quand cette condition est vérifiée on impose la contrainte $\theta_1 = \theta$ en attendant une condition de rupture. La condition de rupture se déduit généralement du comportement qu'aurait le système *en l'absence* de la contrainte.

Contrainte de non-interpénétration (contacts conditionnels)

La contrainte de non-interpénétration des objets est également conditionnelle : soit les objets sont disjoints Γ soit ils vérifient une égalité correspondant à leur contact. Hahn [59] Γ Baraff [9 Γ 10 Γ 11 Γ 12 Γ 13 Γ 14] Γ Moore et Wilhelms [87] introduisent cette contrainte dans le cas de solides rigides. Baraff traite aussi de la complexité du problème consistant à calculer les conditions d'établissement et de rupture des contacts avec ou sans glissement entre n solides rigides. Gascuel [51] gère les collisions et les contacts étendus entre des solides déformables hybrides (squelettes rigides articulés enrobés d'une peau déformable). Li et Canny [75] étudient l'espace libre d'un système de deux solides rigides en contact avec roulement sans glissement. Gourret et Thalmann [57] calculent l'équilibre d'une main serrant une balle déformable Γ modélisées par des éléments finis.

Lois de comportement

Les lois de comportement des solides modélisant leur mouvement Γ leurs déformations et leurs contacts peuvent être considérées comme des contraintes (on pourrait aussi envisager des écoulements de fluides ou de gaz Γ des phénomènes électromagnétiques...).

Les lois de comportement sont dans notre classification des contraintes cinématiques ou dynamiques fixes ; elles peuvent se formuler de façon forte ou faible.

La formulation forte correspond soit à des équations différentielles reliant la dérivée de la quantité de mouvement à la vitesse et à la position Γ ou leurs analogues angulaires Γ soit à des équations aux dérivées partielles reliant la déformation d'un solide à ses champs de contraintes internes et surfaciques.

La formulation faible de ces lois de comportement passe par l'écriture d'un bilan énergétique faisant intervenir les travaux des forces intervenant dans la formulation forte.

Les formulations faible et forte résultant des équations de la mécanique des milieux continus sont présentées dans le chapitre 3 de la partie *Outils*. On utilise plus fréquemment les lois de comportement linéarisées qui conduisent à des équations plus simples.

Des formulations faibles de type “plaques minces” sont utilisées pour la modélisation du comportement de courbes ou de surfaces [24]–[32].

Jouve [69] modélise le comportement du cristallin de l'œil avec une loi non homogène.

Les comportements limites comme la plasticité (déformation permanente de la configuration de référence) la fracture la visco-élasticité (déformation progressive de la configuration de référence) sont également modélisés.

On trouve des références sur la fracture de solides modélisés par des éléments finis dans les revues de méthodes numériques pour l'ingénierie [76]. Dans les publications sur l'animation par ordinateur Norton *et al.* [88] Terzopoulos et Fleischer [103] Desbrun et Gascuel [35] Luciani [80] simulent également de tels comportements.

Contraintes portant sur l'évolution du système

Dans le cas de l'animation les contraintes dépendent également de la variable “temps”. Par exemple : “le point A de l'objet 1 doit se trouver en A_0 à la date t_0 ”.

En effet si les comportements basés sur la physique permettent d'obtenir des animations très réalistes l'animateur doit pouvoir piloter le modèle de façon à réaliser son scénario.

On introduit donc des contraintes semblables à celles que nous venons de voir mais à existence et importance variables en fonction du temps ou de l'état du système lui-même.

L'introduction de ces contraintes se fait le plus souvent par des langages de spécification qui permettent de définir simplement des contraintes de plus en plus évoluées.

2.4 Contrôle

Comme nous l'avons défini plus haut (définition 7) le contrôle est l'algorithme utilisé pour calculer l'état du système qui vérifie le mieux les contraintes dans le cas du modelage ou l'évolution de l'état au cours du temps dans le cas de l'animation.

2.4.1 Contrôle pour le modelage

Dans le cas du modelage de formes le contrôle est un algorithme d'optimisation sous contraintes dont l'inconnue est le vecteur d'état. Les contraintes formulées de façon forte contraignent directement les valeurs que peut prendre le vecteur d'état alors que les contraintes faibles définissent la fonction à optimiser.

A part dans les cas simples il est nécessaire de recourir à des méthodes d'optimisation itératives présentées dans les ouvrages de Ciarlet [29] et Minoux [83].

La méthode des éléments finis [116-117-115] qui permet de ramener un problème d'optimisation à la résolution d'un système d'équations (linéaires ou non linéaires) peut aussi être utilisée.

2.4.2 Contrôle pour l'animation

En animation le contrôle consiste à intégrer les équations différentielles modélisant le comportement des solides.

La méthode d'intégration la plus souvent utilisée est la méthode d'Euler qui permet de calculer directement l'état du système à la date $t + \Delta t$ connaissant son état à la date t en évaluant une seule fois la dérivée de l'état (les forces et les moments). La méthode d'Euler présente l'inconvénient de n'être que d'ordre 1 (voir [106-95]) c'est à dire que l'erreur faite en calculant l'état à la date $t + \Delta t$ est de l'ordre de Δt .

En abandonnant l'avantage de n'avoir à calculer qu'une seule fois la dérivée du système la méthode "d'Euler améliorée" permet d'atteindre l'ordre 2 (erreur de l'ordre de Δt^2) en évaluant deux fois la dérivée à chaque pas. Si l'on accepte d'évaluer quatre fois la dérivée de l'état du système à chaque pas on peut utiliser la méthode de Runge-Kutta d'ordre 4. Il existe d'autres méthodes encore plus performantes rendant nécessaire la mémorisation d'un plus grand nombre d'états du système (voir les deux références citées plus haut).

Il est à noter que l'utilisation de méthodes d'ordre élevé permet à précision égale d'augmenter le pas d'intégration Δt . Le gain en temps de calcul compense alors largement le surcoût des évaluations multiples.

L'introduction des contraintes se fait le plus fréquemment en ajoutant des forces artificielles favorisant leur réalisation. Ces forces font naturellement osciller le système autour de la solution.

Dans le cas des contraintes de distance entre particules vues plus haut Lombardo [79] propose d'éliminer les oscillations en aplatissant localement la cuvette d'énergie potentielle au voisinage du minimum. Généralement les oscillations sont atténuées en ajoutant des forces de "frottement" artificielles éliminant de l'énergie cinétique.

Récemment plusieurs auteurs ont proposé un traitement hybride des contraintes géométriques: en plus des forces attirant le système vers des états vérifiant les contraintes l'état du système est modifié directement par petits "sauts" vers le plus proche état vérifiant les contraintes. Baraff [14] utilise cette technique pour simuler l'évolution d'un système de solides rigides en contact (non-interpénétration). Gascuel et Gascuel [49] l'utilisent pour vérifier des contraintes de type "point sur surface" dans un système de solides déformables.

Cette méthode hybride permet quand l'amplitude des "sauts" n'est pas trop importante de vérifier la contrainte en éliminant les oscillations. L'inconvénient de la méthode est la perte de l'énergie représentée par ces déplacements qui n'a pas de justification physique.

2.5 Conclusion

Dans ce chapitre bibliographique nous avons exposé les travaux existants dans les domaines de l’animation et du modelage de formes par ordinateur.

Nous avons dégagé trois composantes communes aux systèmes d’animation et de modelage : la représentation de la déformation et de la géométrie, les contraintes et l’algorithme de contrôle.

Pour chacune de ces composantes nous avons présenté les différents choix adoptés dans la littérature.

Le problème de la simulation interactive du comportement des organes et des instruments au cours d’un geste chirurgical fait actuellement l’objet d’un nombre croissant de recherches.

En effet, la puissance des stations de travail graphiques et l’apparition de capteurs de position tridimensionnels permettent d’envisager à court terme l’apparition de simulateurs chirurgicaux dans les blocs opératoires.

Outre la réalisation d’un modèle suffisamment robuste et réaliste, les applications médicales posent deux autres problèmes difficiles :

- Reconstruire ou segmenter les surfaces des organes afin de fournir les géométries initiales des solides du système. La reconstruction peut se faire à partir d’images issues de modalités différentes telles que la radiographie, la tomographie, la tomographie par émission de positrons, l’imagerie par résonance magnétique... L’utilisation d’atlas anatomiques pour injecter de la connaissance dans le processus de segmentation devrait permettre de l’accélérer et de résoudre certains cas délicats comme la séparation des os aux articulations ou celle des muscles voisins.
- Mesurer ou estimer les paramètres des lois de comportement liés aux organes comme l’élasticité et la résistance des tissus. Quelques travaux existent dans le cas des muscles [28] et de l’œil [69]. On peut envisager deux solutions à ce problème. D’une part, disposer des capteurs de contraintes sur les instruments chirurgicaux et établir des statistiques sur les mesures effectuées pendant les opérations. D’autre part, utiliser l’expertise des chirurgiens pour ajuster les paramètres dans un simulateur en fonction de leurs sensations.

Dans le chapitre suivant nous voyons les caractéristiques que doit présenter un modèle “chirurgical” ainsi que différentes expériences d’implémentation et de comparaison des choix possibles concernant la géométrie et les déformations, les contraintes et le contrôle.

Chapitre 3

Un modèle pour la simulation d'un geste chirurgical

La simulation du comportement d'organes lors d'un geste chirurgical est un cas particulier d'animation interactive de systèmes de solides. Dans ce cas la nature du système impose certaines contraintes au modèle mais permet aussi des hypothèses simplificatrices. Dans ce contexte décrit dans la première section il s'avère qu'aucun des modèles existants ne correspond aux spécifications.

Il est donc nécessaire de concevoir un nouveau modèle. Des études dans ce sens sont exposées dans ce chapitre. Pour ce faire nous adoptons le même découpage qu'au chapitre précédent : déformations et géométrie, contraintes, contrôle. Nos expériences ont pour point de départ :

- les modèles existants présentés au chapitre précédent
- la mécanique des milieux continus et la modélisation des comportements élastiques résumés dans le chapitre 3 de la partie *Outils*
- les méthodes d'éléments finis présentées succinctement dans le chapitre 3 de la partie *Outils* et que nous abordons d'un point de vue pratique dans ce chapitre.

3.1 Contexte et idées

Le système à simuler est constitué de solides très déformables (organes, vaisseaux) et rigides (os, instruments chirurgicaux). Le modèle doit donc permettre la **cohabitation de solides très déformables et de solides rigides**. Dans les modèles de type masses/ressorts la modélisation de solides rigides conduit à des ressorts de raideur infinie et donc à une grande instabilité des algorithmes de contrôle.

Les organes et vaisseaux sont lisses (sans points anguleux) et entretiennent des **contacts étendus permanents** les uns avec les autres. On peut supposer que ces contacts sont sans friction. L'algorithme de contrôle doit détecter et prendre en compte ces contacts étendus. On doit gérer une représentation explicite de l'ensemble des surfaces en contact.

Les instruments sont rigides ou rigides articulés. Certains de ces instruments permettent d'inciser les tissus : la résistance des tissus puis leur déchirement doivent

être pris en compte. En particulier, la **topologie des objets** doit pouvoir changer au cours de l'évolution du système (création de trous).

L'affichage des objets doit se faire en temps réel. La position de certains instruments du système peut être couplée aux positions d'instruments réels mesurées par des localisateurs tridimensionnels. Dans le cas où l'on souhaite fournir une **interface gestuelle** au simulateur, le modèle physique et l'algorithme de contrôle devront être suffisamment robustes et proches de la réalité physique.

Il n'existe pas de modèle vérifiant toutes les conditions énoncées ci-dessus. La conception d'un tel modèle est l'objet du reste de ce chapitre. Ces travaux sont à rapprocher de ceux de Baraff, Witkin et Gascuel pour la gestion des contacts permanents, de Terzopoulos, Metaxas, Gourret et Thalmann pour la prise en compte des déformations, de Terzopoulos, Szeliski, Tonnesen, Delingette, Subsol, Pignon et Cotin pour la gestion des changements "chirurgicaux" de la géométrie (les références à ces travaux sont données dans le chapitre 2).

Une des idées clés de notre approche est de faire l'hypothèse qu'à chaque instant, le système est dans un état d'équilibre ; on parle d'**évolution quasi statique**. Dans ce cas, il n'est pas nécessaire de prendre en compte les accélérations et les vitesses des objets.

Cette hypothèse a plusieurs avantages : la diminution du nombre de variables, la suppression des oscillations qui apparaissent dans les systèmes dynamiques au voisinage de l'équilibre, la simplification des équations correspondant aux contraintes de contact par rapport aux travaux de Baraff [9, 10, 11] dans des systèmes de solides rigides, repris par Witkin et lui-même [8] dans le cas des solides déformables.

Enfin, même en l'absence des vitesses et accélérations des solides, les déformations permettent de calculer les champs de contraintes présents sur la surface des objets, ce qui rend possible l'utilisation de périphériques de retour d'effort.

Nous choisissons d'associer une **énergie potentielle** à chaque état du système. Cette énergie prend en compte l'énergie potentielle de pesanteur de tous les solides et l'énergie potentielle élastique des solides déformables. L'énergie potentielle des solides déformables est définie en intégrant une densité d'énergie potentielle sur l'intérieur de chaque solide (hyperélasticité).

L'algorithme de contrôle consiste alors à maintenir à chaque instant le système dans un **minimum local** de cette énergie à l'intérieur de l'espace libre.

3.2 Géométrie et déformations

Lors de nos expériences, nous avons utilisé quatre représentations de la géométrie et des déformations des solides. D'une part, nous avons repris trois représentations des solides déformables (exposées au chapitre 2) : les systèmes de particules orientées, les déformations globales et les maillages de type éléments finis. D'autre part, nous introduisons aussi dans le système des solides rigides polyédriques. Les représentations déformables serviront à modéliser les organes mous et les solides rigides seront utilisés pour les os et les instruments chirurgicaux.

Les représentations peuvent cohabiter au sein du même système, ce qui répond à l'une des exigences vues plus haut.

Nous n'exposons ici que les résultats concernant la représentation par des maillages de type éléments finis et les solides rigides polyédriques.

En effet le calcul de l'énergie élastique d'un solide représenté par des particules orientées nécessite en pratique soit le calcul des “voisinages” des particules soit la détermination globale de la déformation définie par la position des particules. Dans le premier cas on retrouve un maillage de type éléments finis et dans le second une déformation globale.

Des approches simplifiées utilisant les systèmes de particules et les déformations globales pour la simulation chirurgicale sont proposées dans [66-32].

Afin de bien séparer la représentation de la géométrie et des déformations des autres composantes du modèle nous avons choisi de considérer chaque objet comme une **boîte noire** offrant un certain nombre de fonctionnalités.

La liste des fonctionnalités donnée ci-après résulte de nos expériences. Nous la donnons maintenant afin d'augmenter la clarté de l'exposé. Nous verrons ensuite comment les réaliser dans chacune des représentations envisagées.

Cette “approche objet” du système peut être directement implémentée dans des langages objets à classes tels que le **C++**.

3.2.1 Fonctionnalités

Dans ce qui suit le terme *objet* désigne la structure de données associée à un solide. On considère uniquement les solides pour lesquelles les grandeurs utilisées dans ce chapitre ont un sens. En particulier leur intérieur devra être non vide et connexe leur frontière orientable et dérivable deux fois presque partout.

Les fonctionnalités offertes par le solide du point de vue de sa géométrie et de sa déformation concernent trois points (figure 17) : le paramétrage externe le calcul d'une énergie potentielle associée à la déformation et l'accès à la frontière du solide déformé.

Paramétrage de la déformation

La déformation d'un solide est paramétrée par un vecteur a . Ce vecteur contient toutes les variables permettant au solide de connaître sa déformation.

Alors que dans les modèles usuels ces paramètres sont gérés en interne par chaque objet nous avons choisi de les conserver à l'**extérieur** de l'objet : ce sont les algorithmes manipulant le solide qui lui communiquent les valeurs de a . De cette façon les fonctions de contrôle peuvent modifier a sans avoir à faire partie de l'objet.

Chaque objet possède une fonction mise à disposition de ses utilisateurs et donnant la dimension de son vecteur de paramètres.

D'autre part puisque l'utilisateur ne connaît pas la relation entre les composantes de a et la déformation du solide (cette relation est connue uniquement dans l'objet représentant le solide) l'objet doit aussi fournir des fonctions d'initialisation des composantes de a et d'application de transformations géométriques de base (translation rotation).

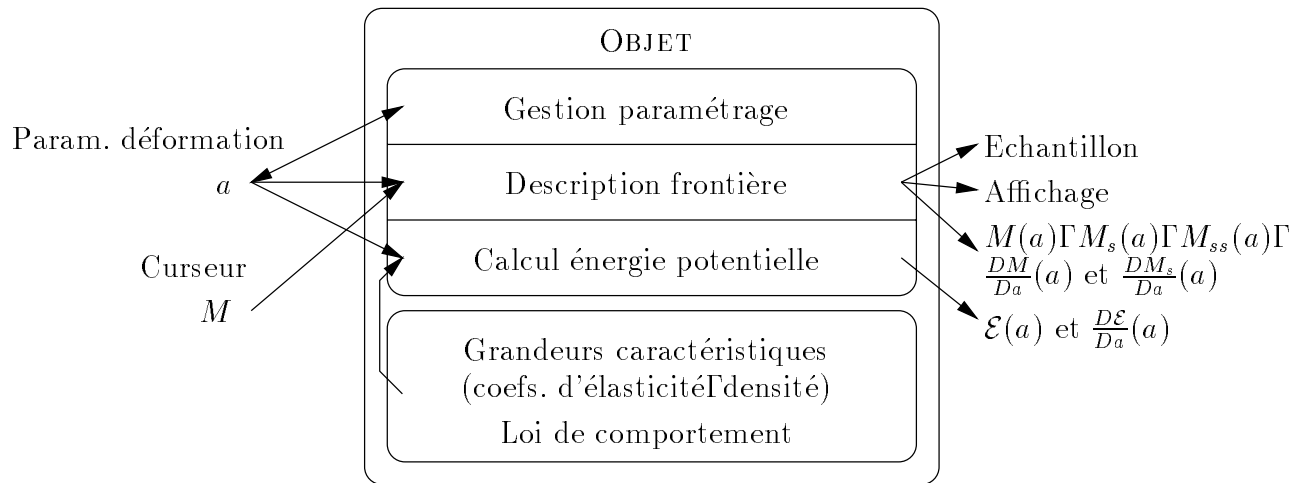


Fig. 17 - Fonctionnalités d'un objet associé à un solide, indépendamment de la représentation sous-jacente de la géométrie et des déformations.

Energie potentielle

Une fonction de l'objet permet de calculer son énergie potentielle $\mathcal{E}(a)$ dans la déformation de paramètre a . $\mathcal{E}(a)$ est la somme d'un terme d'énergie potentielle élastique $W(a)$ et d'un terme d'énergie potentielle de pesanteur $G(a)$.

$$\mathcal{E}(a) = W(a) + G(a). \quad (3.1)$$

Cette fonction renvoie aussi le gradient de l'énergie potentielle en fonction de a ce qui permet de mettre en œuvre des algorithmes d'optimisation pour contrôler l'évolution du système.

W dépend d'une loi de comportement renvoyant la densité d'énergie potentielle et son gradient en fonction du tenseur des déformations de Green-Lagrange E (voir chapitre 3 partie *Outils*). G dépend de la densité volumique du solide et de l'accélération de la pesanteur g .

Frontière

La frontière du solide a besoin d'être connue pour trois opérations : l'affichage, la détection des intersections et la prise en compte des contraintes de contact. Les fonctions d'accès à la frontière que nous voyons dans ce paragraphe concernent le solide dans une configuration quelconque et dépendent donc du paramètre a .

Pour l'affichage, une fonction permet d'envoyer un ensemble de primitives graphiques décrivant la surface du solide dans la configuration courante à une librairie graphique (X11, PEX, OpenGL, génération de PostScript, ..).

La détection des intersections nécessite un échantillon de la frontière sous la forme d'un nuage de points uniformément répartis et de densité connue. En effet, nous utilisons la méthode "par points" implémentée en 1992 dans notre modèle des

δ -snakes [473] pour détecter les intersections Γ plutôt que les méthodes “par facettes” généralement employées. Nous détaillons cet algorithme de détection dans la section 3.4.

La prise en compte des contraintes nécessite la connaissance de points particuliers sur les contours des objets : points de contact Γ points dont la position est fixée. La représentation de points sur les frontières se fait à travers la notion de **curseur**.

Un curseur présente les mêmes fonctionnalités quelle que soit la représentation de la géométrie et des déformations de l’objet auquel il appartient : l’algorithme de contrôle interroge les curseurs Γ qui font à leur tour appel à leur objet pour répondre à ces requêtes Γ de façon transparente pour l’algorithme de contrôle.

Les fonctionnalités que doivent offrir les curseurs sont dictées par le choix de l’algorithme de contrôle : un curseur doit pouvoir être **déplacé** le long du contour sur lequel il se trouve Γ et donner les **propriétés différentielles locales** (tangente Γ dérivée seconde Γ dérivée selon a) au point du contour correspondant à sa position.

Nous donnons dans cette section les détails de leur implémentation selon les représentations. Les fonctionnalités des curseurs seront exploitées dans la section 3.4 consacrée au contrôle.

Le passage à la dimension 3 introduit des barrières fondamentales à la définition de paramétrages globaux réguliers des surfaces ; notre choix de n’utiliser que des propriétés locales pour les curseurs permet de contourner cette difficulté.

Comportements limites

La prise en compte des comportements limites Γ impliquant une déformation de la géométrie de référence Γ dépend de la déformation représentée par a et de conditions de rupture.

Ces comportements ne sont pas traités dans notre exposé.

3.2.2 Maillages de type éléments finis

Nous allons maintenant voir comment représenter un solide par un maillage de type “éléments finis” et comment implémenter les fonctionnalités que nous venons de décrire dans ce cas.

Les méthodes d’éléments finis sont utilisées depuis des dizaines d’années (voir [116]) dans tous les domaines des sciences de l’ingénieur. Elles servent principalement à calculer des champs régis par des équations aux dérivées partielles : déformations et contraintes dans des structures (bâtiments Γ barrages Γ pièces mécaniques) Γ écoulements de fluides Γ champs électro-magnétiques.

Bien que les méthodes d’éléments finis permettent également de calculer des champs de déformations et de contraintes dépendant du temps [15] Γ elles n’ont été que peu utilisées pour l’animation et le modelage de formes par ordinateur.

Nous allons présenter en détail un exemple de modélisation d’un objet déformable par des éléments paramétriques quadratiques en dimension $d = 2$ sachant que le principe reste le même pour tous les types d’éléments Γ en dimensions 2 et 3.

L’intérieur Ω du solide est pavé par des éléments paramétriques quadratiques à quatre côtés.

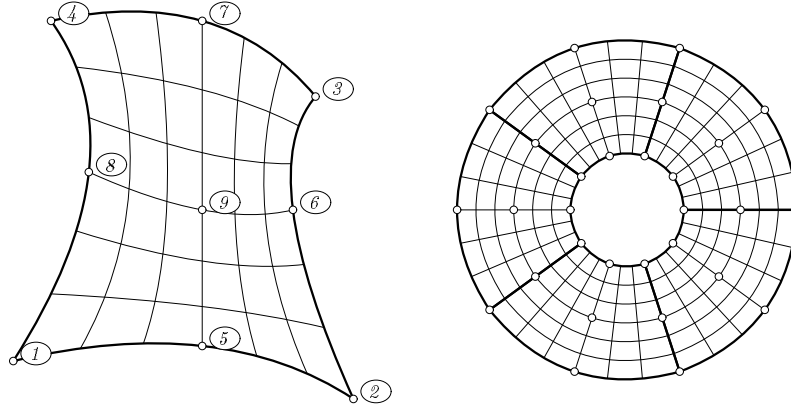


Fig. 18 - (a) *Elément paramétrique quadratique à quatre côtés et 9 nœuds.* (b) *Solide “anneau” constitué de cinq éléments de ce type.*

La déformation est **interpolée** à l’intérieur de chaque élément en fonction des positions des **nœuds** définissant l’élément. Le recollement des interpolations sur chaque élément permet d’obtenir une déformation définie sur Ω . Les paramètres de cette déformation sont donc les coordonnées de tous les nœuds du pavage.

Elément

Les éléments paramétriques sont décrits dans la plupart des ouvrages sur les méthodes d’éléments finis. On pourra par exemple se référer à Bathe dans [15]. L’élément de la figure 18a est défini par un ensemble N de 9 nœuds (numérotés de 1 à 9 sur la figure).

Deux points sont associés à chaque nœud $n \in N$: R_n représentant la position du nœud n dans la configuration de référence du solide et C_n sa position dans la configuration courante.

La position d’un point de l’élément est donnée par un couple de coordonnées locales (u, v) dans le carré $[-1, 1]^2$. A ce couple correspond un vecteur $P(u, v)$ de 9 poids. Le point $R(u, v)$ de la configuration de référence de coordonnées locales (u, v) est le barycentre des points R_n avec les coefficients $P(u, v)_n$ (on indice $P(u, v)$ par $n \in N$) et de même pour $C(u, v)$ le point correspondant aux coordonnées locales (u, v) dans la configuration courante :

$$\begin{aligned} R(u, v) &= \sum_{n \in N} P(u, v)_n R_n, \\ C(u, v) &= \sum_{n \in N} P(u, v)_n C_n. \end{aligned} \quad (3.2)$$

Afin d’avoir une écriture matricielle on notera \overline{R} la matrice $d \times |N|$ formée des $|N|$ vecteurs $R_n \in \mathbb{R}^d$ et de même pour \overline{C} ; on peut alors écrire matriciellement les

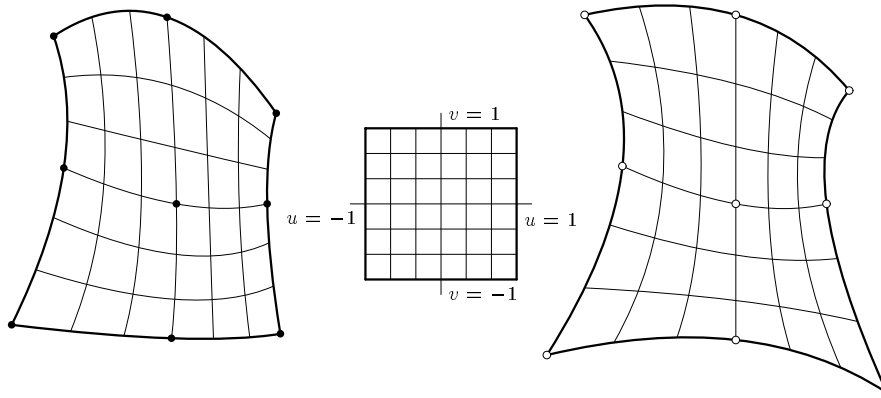


Fig. 19 - Les trois systèmes de coordonnées associés à un élément. (a) Configuration de référence, les points R_n sont dessinés en noir. (b) Coordonnées locales de l'élément. (c) Configuration courante, les points C_n sont dessinés en blanc.

relations précédentes :

$$\begin{aligned} R(u, v) &= \overline{R}P(u, v), \\ C(u, v) &= \overline{C}P(u, v). \end{aligned} \quad (3.3)$$

On voit que de cette façon on importe quelle grandeur définie aux nœuds peut être interpolée à l'intérieur de l'élément.

Les fonctions de pondération $P(u, v)_n$ (ou fonctions de forme) sont des polynômes en u et v nuls en chaque nœud sauf au nœud n où elles valent 1. La figure 20 montre ces fonctions de pondération (il y en a trois types pour cet élément).

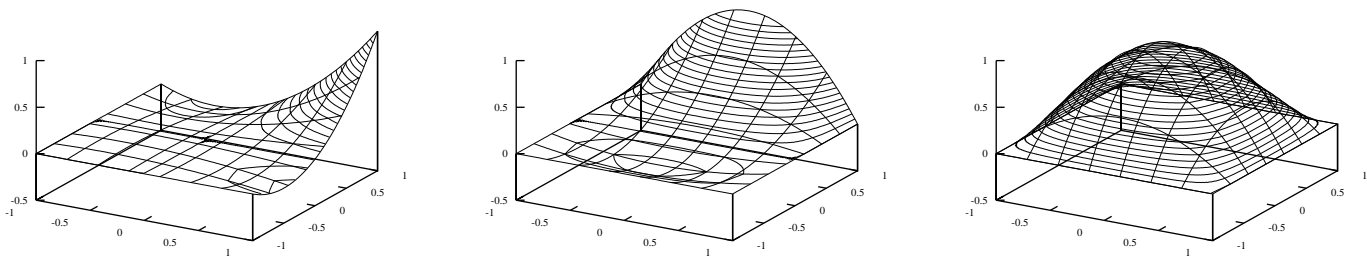


Fig. 20 - Fonctions de pondération des éléments quadratiques à 9 nœuds. (a) Nœuds 1-4. (b) Nœuds 5-8. (c) Nœud 9.

Il existe une immense variété d'éléments selon le nombre et la position des nœuds définissant la fonction d'interpolation la nature de cette fonction la dimension de

l'espace Γ celle de l'élément...; il existe aussi des éléments adaptés à la géométrie de l'objet : infinis Γ de révolution Γ périodiques Γ à trous... Le principe d'interpolation reste néanmoins le même dans tous les cas.

Assemblage d'éléments

Dans le cas d'un solide pavé par plusieurs éléments Γ les nœuds sont partagés entre les éléments Γ ce qui assure la continuité des fonctions interpolées sur l'intérieur du solide. Il faut toutefois éviter le problème des recollements non compatibles Γ illustré par la figure 21.

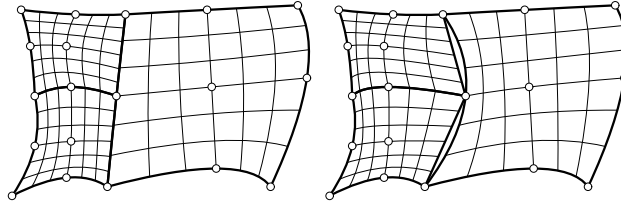


Fig. 21 - Problèmes de recollement pouvant apparaître entre des frontières de degrés différents. (a) Recollement correct. (b) Recollement discontinu.

Le vecteur a paramètre de la déformation courante est l'ensemble des coordonnées des points C_n pour tous les nœuds du solide. a est donc de dimension d fois le nombre total de nœuds. Ainsi Γ le vecteur de paramètres de l'anneau (figure 18b) Γ comportant 30 nœuds Γ est de dimension 60.

Un objet représenté de cette façon contient un ensemble de nœuds Γ un ensemble d'éléments Γ et un ensemble de contours. Chaque élément connaît les nœuds servant à le paramétrer. Un contour est une suite d'arcs définis par des nœuds du maillage ; dans le cas exposé ici Γ ces arcs sont des courbes paramétriques de degré 2.

Gradient de la déformation

Nous avons vu qu'à un point $U = (u, v)$ du carré $[-1, 1]^2$ correspondent deux points : $R(U)$ dans la configuration de référence et $C(U)$ dans la configuration courante. La déformation φ est donc définie implicitement sur $R([-1, 1]^2)$ par :

$$\varphi(R(U)) = C(U). \quad (3.4)$$

Cette relation permet de déduire le gradient de la déformation en $R(U)$:

$$\nabla \varphi(R(U)) = \left(\overline{C} \frac{DP}{DU}(U) \right) \left(\overline{R} \frac{DP}{DU}(U) \right)^{-1}. \quad (3.5)$$

$\frac{DP}{DU}(U)$ est la matrice de d colonnes de $|N|$ lignes dont la k -ième colonne est le vecteur $\frac{dP}{dU_k}(U)$.

La figure 22 montre le champ du gradient sur l'anneau dans quatre déformations différentes.

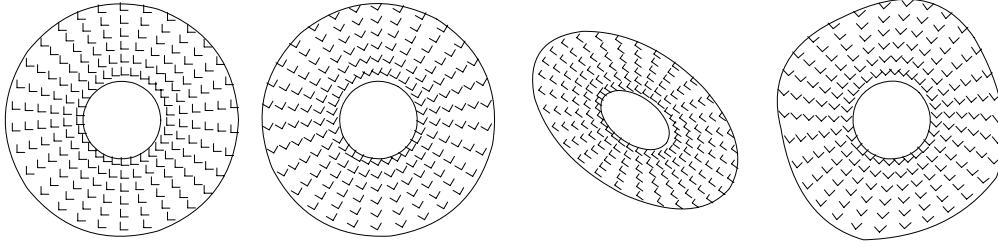


Fig. 22 - Champs de gradient dans l'anneau sous plusieurs déformations. (a) Identité. (b) Rotation d'angle $\pi/3$. (c) Affinité d'axe x et de rapport 0.6 suivie d'une rotation d'angle $-\pi/5$. (d) Transformation quelconque.

Si le gradient doit être évalué un grand nombre de fois au même point U On voit qu'il est intéressant de mémoriser une fois pour toutes la matrice $\frac{DP}{DU}(U) \left(\overline{R} \frac{DP}{DU}(U)\right)^{-1}$. C'est par exemple le cas aux points d'intégration de la densité d'énergie (paragraphe suivant).

Energie potentielle

Si $w(E)$ est la densité d'énergie élastique (i.e. l'énergie potentielle élastique par unité de volume dans la configuration de référence voir [30] section 1.4) associée au tenseur des déformations de Green–Lagrange $E = \frac{1}{2}(\nabla\varphi^T\nabla\varphi - I)$ $\Gamma\mu$ la masse volumique du solide dans la configuration de référence Γ et g l'accélération de la pesanteur On peut calculer l'énergie potentielle $\mathcal{E}_e(a)$ associée à la portion du solide couverte par un élément e :

$$\mathcal{E}_e(a) = \int_{R([-1,1]^2)} w(E(R)) - \mu \langle g, C(R) \rangle dR. \quad (3.6)$$

Cette intégrale est calculée sur l'image de l'élément dans la configuration de référence ; nous la transportons par changement de variable dans l'espace des coordonnées locales U :

$$\mathcal{E}_e = \int_{[-1,1]^2} (w(E(U)) - \mu \langle g, C(U) \rangle) \det\left(\overline{R} \frac{DP}{DU}(U)\right) dU. \quad (3.7)$$

En effet On dispose (voir [116] sections 8.8 à 8.11) de méthodes d'intégration numérique d'une fonction définie sur $[-1,1]^2$. Il s'agit des méthodes de quadrature de Gauss dans lesquelles l'intégrale est approchée par une somme pondérée des valeurs de la fonction en des points $x_p \in [-1,1]^2$:

$$\int_{[-1,1]^2} f(U) dU \approx \sum_p H_p f(x_p). \quad (3.8)$$

Les poids H_p et les points x_p sont choisis de telle sorte que l'approximation soit exacte pour des fonctions polynomiales des composantes de U le degré de ces fonctions étant maximal pour un nombre de points donné.

La figure 23 montre les points d'évaluation x_p de la fonction à intégrer et leur poids H_p pour 4, 9 et 16 points.

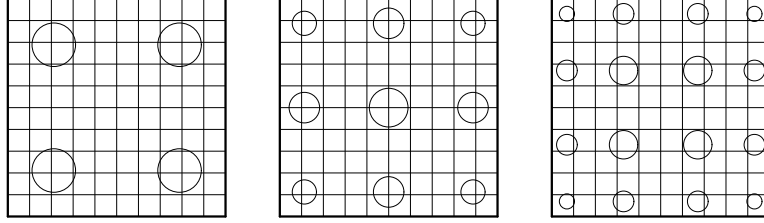


Fig. 23 - Points d'évaluation de la fonction dans les méthodes de Gauss à 4, 9 et 16 points. Ces points sont les centres des cercles, dont la surface est proportionnelle au poids du point dans l'approximation de l'intégrale de la fonction.

L'énergie potentielle totale du solide est la somme des énergies de chaque élément :

$$\mathcal{E}(a) = \sum_e \mathcal{E}_e(a). \quad (3.9)$$

Les indications données dans la suite de ce paragraphe traitent du calcul de E et du gradient de $w(E)$ selon a .

Du point de vue des performances il est intéressant de mémoriser les quantités ne dépendant pas de a en chacun des points d'évaluation de la densité d'énergie. En un point d'évaluation de coordonnées locales U et de poids H on peut calculer :

- la matrice $Q(U) = \frac{DP}{DU}(U) \left(\overline{R} \frac{DP}{DU}(U) \right)^{-1} \Gamma$
- le réel $H \det(\overline{R} \frac{DP}{DU}(U)) \Gamma$
- le vecteur $P(U)$.

En dimension 2 E étant symétrique $w(E)$ ne dépend que de E_{11} , E_{22} et E_{21} . La fonction donnant w donnera aussi les dérivées partielles

$$\frac{\partial w}{\partial E_{11}}, \quad \frac{\partial w}{\partial E_{22}}, \quad \frac{\partial w}{\partial E_{21}}. \quad (3.10)$$

On a d'autre part en notant $J = \nabla \varphi$ pour alléger l'écriture $E = \frac{1}{2}(J^T J - I)$ soit :

$$\begin{aligned} E_{11} &= \frac{1}{2}(J_{11}^2 + J_{21}^2 - 1), \\ E_{22} &= \frac{1}{2}(J_{12}^2 + J_{22}^2 - 1), \\ E_{21} &= \frac{1}{2}(J_{11}J_{12} + J_{21}J_{22}). \end{aligned} \quad (3.11)$$

On en déduit les dérivées partielles de w en fonction des coefficients de J :

$$\begin{aligned}\frac{\partial w}{\partial J_{11}} &= J_{11} \frac{\partial w}{\partial E_{11}} + \frac{1}{2} J_{12} \frac{\partial w}{\partial E_{21}}, \\ \frac{\partial w}{\partial J_{21}} &= J_{21} \frac{\partial w}{\partial E_{11}} + \frac{1}{2} J_{22} \frac{\partial w}{\partial E_{21}}, \\ \frac{\partial w}{\partial J_{12}} &= J_{12} \frac{\partial w}{\partial E_{22}} + \frac{1}{2} J_{11} \frac{\partial w}{\partial E_{21}}, \\ \frac{\partial w}{\partial J_{22}} &= J_{22} \frac{\partial w}{\partial E_{22}} + \frac{1}{2} J_{21} \frac{\partial w}{\partial E_{21}}.\end{aligned}\quad (3.12)$$

On peut maintenant calculer les dérivées partielles de w selon les coefficients de $\overline{C}\Gamma$ et donc Γ à renumérotation près Γ selon les coefficients de a :

$$\frac{\partial w}{\partial C_{in}} = Q_{n1} \frac{\partial w}{\partial J_{i1}} + Q_{n2} \frac{\partial w}{\partial J_{i2}}. \quad (3.13)$$

La figure 24 montre la densité d'énergie élastique dans l'anneau déformé.

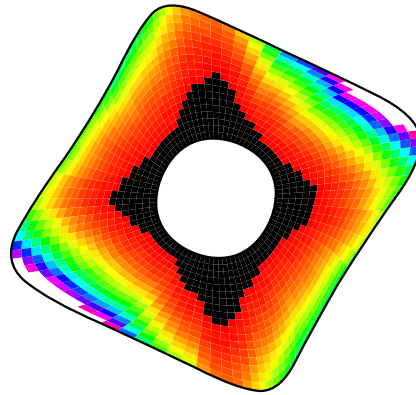


Fig. 24 - Densité d'énergie élastique dans l'anneau déformé.

Frontière

Dans le cas des solides représentés par les éléments quadratiques à 9 nœuds définis plus haut Γ les contours sont constitués d'arcs de courbes paramétriques de degré 2 et de dimension 1 Γ comme celui représenté figure 25.

La position d'un point de cet arc est la restriction de l'interpolation définie plus haut sur l'élément à l'arc. Il s'avère qu'elle ne dépend que de la position des trois sommets sur lesquels elle s'appuie et est **indépendante** de celle des six autres sommets de l'élément.

Si on paramètre l'arc par un réel $u \in [-1, 1]$ Γ les positions $R(u)$ et $C(u)$ du point d'abscisse u respectivement dans la configuration de référence et dans la configura-

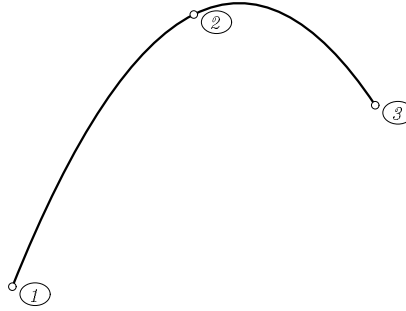


Fig. 25 - Arc de courbe paramétrique polynomiale de degré 2 constituant les contours des solides. Le point 1 correspond à $u = -1$, le point 2 à $u = 0$ et le point 3 à $u = +1$.

tion déformée sont définies en fonction des positions des trois nœuds par :

$$\begin{aligned} R(u) &= P_1(u)R_1 + P_2(u)R_2 + P_3(u)R_3, \\ C(u) &= P_1(u)C_1 + P_2(u)C_2 + P_3(u)C_3, \end{aligned} \quad (3.14)$$

avec les polynômes d'interpolation :

$$P_1(u) = \frac{1}{2}u(u-1), \quad P_2(u) = (1-u^2), \quad P_3(u) = \frac{1}{2}u(u+1). \quad (3.15)$$

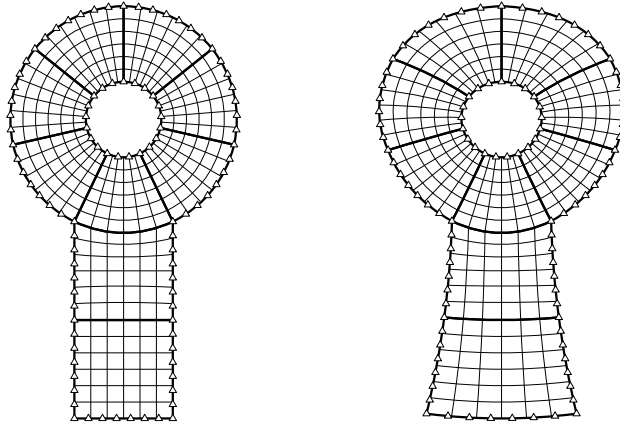


Fig. 26 - Echantillons des contours de l'objet "clé". (a) Dans la configuration de référence. (b) Dans une configuration déformée.

Nous avons choisi de paramétrer localement le contour par son abscisse curviligne s dans la configuration de référence. Ce choix garantit la continuité C^1 du paramétrage sur deux arcs voisins ayant deux dérivées colinéaires à leur jonction.

Dans ce cas les formules donnant le comportement local en un curseur M du contour déformé Γ correspondant au point d'abscisse locale u de l'arc sont les suivantes en posant $\lambda = \|R_u\|^{-1}$:

$$\begin{aligned} M &= C, & M_s &= \lambda C_u, \\ M_{ss} &= \lambda^2 C_{uu} - \lambda^4 \langle R_u, R_{uu} \rangle C_u, \\ \frac{\partial M_x}{\partial C_{kx}} &= \frac{\partial M_y}{\partial C_{ky}} = P_k(u), & \frac{\partial M_{sx}}{\partial C_{kx}} &= \frac{\partial M_{sy}}{\partial C_{ky}} = P'_k(u). \end{aligned} \quad (3.16)$$

Les dérivées de C et R selon u s'obtiennent facilement en dérivant les polynômes P_k dans 3.14. La figure 27 montre les dérivées selon s sur la frontière de l'objet "clé" dans une configuration déformée.

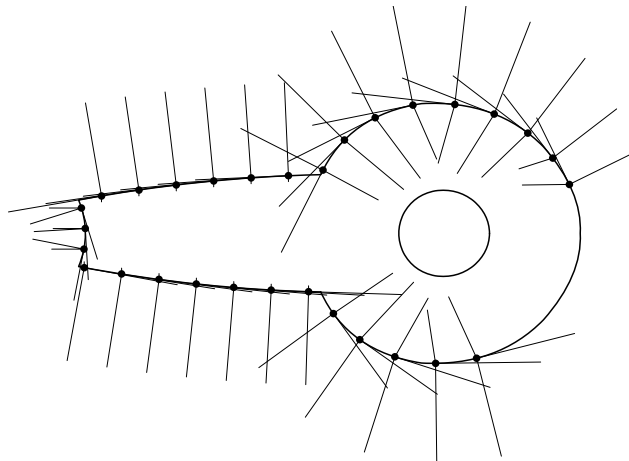


Fig. 27 - Dérivée, normale et dérivée seconde selon s pour un curseur se déplaçant sur le contour extérieur de l'objet "clé" déformé.

3.2.3 Solides rigides polyédriques

En dimension 2 l'on parlera de solides rigides polygonaux. Cette représentation est beaucoup plus simple que la précédente. Même si l'on peut représenter des solides rigides avec une représentation gérant les solides déformables il est bien plus avantageux d'utiliser une représentation spécifique.

La déformation est rigide et le vecteur associé a ne possède donc que 3 paramètres en dimension 2 et 6 en dimension 3.

L'énergie potentielle ne comporte que la partie due au poids qui ne dépend que de l'**altitude** du centre de gravité du solide $G(a)$; si m est la masse du solide et g l'accélération de la pesanteur l'on a :

$$\mathcal{E}(a) = -m \langle g, G(a) \rangle. \quad (3.17)$$

La frontière de ces objets est constituée de segments qui constituent un cas particulier des arcs paramétriques vus dans la section précédente. Il n'est donc pas nécessaire de donner des détails sur l'implémentation de la structure de curseur dans ce cas.

3.3 Contraintes

La déformation de chaque solide i est paramétrée par un vecteur d'état a_i (que nous avons noté a dans la section précédente afin de ne pas alourdir les notations). L'ensemble du système est donc paramétré par un "grand" vecteur \mathbf{A} (concaténation de tous les vecteurs a_i).

Chaque contrainte géométrique se traduit par une condition sur le vecteur d'état global du système \mathbf{A} .

Comme nous l'avons défini au chapitre 2 les contraintes formulées de façon faible introduisent un terme $L(\mathbf{A})$ dans la fonctionnelle d'énergie du système. Nous avons aussi besoin de calculer le gradient de $L(\mathbf{A})$ selon \mathbf{A} pour utiliser l'algorithme d'optimisation.

Les contraintes fortes imposent une ou plusieurs équations portant sur la variation $\Delta\mathbf{A}$ du vecteur d'état du type $\langle \Delta\mathbf{A}, U \rangle = l$ pour les égalités et $\langle \Delta\mathbf{A}, U \rangle \geq l$ pour les inégalités.

Imaginons la contrainte matérialisée dans l'espace des états par une surface : les états se trouvant sur la surface vérifient la contrainte. Dans le cas d'une égalité seuls les états se trouvant sur la surface sont autorisés alors que dans le cas d'une inégalité l'espace libre se trouve d'un côté de la surface et l'autre côté est interdit.

Pour un état \mathbf{A} proche de la surface matérialisant une contrainte on ne s'intéresse qu'à l'aspect local de la surface : quel est son plan tangent et à quelle distance en sommes-nous ? Ces informations sont représentées par U et l .

Nous introduisons quatre contraintes : la liaison fixe entre un point d'un solide et un point fixe du plan, la liaison fixe entre deux points de deux solides, le contact entre un solide et un demi-plan et enfin le contact entre deux solides (qui concerne aussi le contact d'un solide avec lui-même).

Pour les deux contraintes fixes nous donnons les formulations forte et faible dans le paragraphe 3.3.1. Pour les contraintes de contact nous donnons la formulation forte et les conditions d'établissement et de rupture de la contrainte dans le paragraphe 3.3.2.

3.3.1 Liaisons

Liaison d'un point d'un solide avec un point fixé

On impose au curseur M fixe sur la frontière d'un solide d'être constamment en B . La position de M ne dépend donc que de \mathbf{A} . Le coefficient correspondant à la formulation faible est :

$$L(\mathbf{A}) = \|M - B\|, \quad \frac{DL}{D\mathbf{A}}(\mathbf{A}) = \frac{1}{\|M - B\|} \left\langle \frac{DM}{D\mathbf{A}}, M - B \right\rangle. \quad (3.18)$$

La formulation forte de cette contrainte s'écrit :

$$\left\langle \Delta\mathbf{A}, \frac{DM^T}{D\mathbf{A}}(M - B) \right\rangle = -\|M - B\|^2. \quad (3.19)$$

On l'obtient en substituant une approximation de $\|M(\mathbf{A}) - B\|$ au premier ordre selon \mathbf{A} dans $\|(M(\mathbf{A} + \Delta\mathbf{A}) - B)\| = 0$.

Liaison point–point entre solides

On impose aux deux curseurs M_1 et M_2 fixes sur les frontières de leurs solides d’occuper constamment la même position. On définit de la même façon que précédemment le coefficient :

$$\begin{aligned} L(\mathbf{A}) &= \|M_1 - M_2\|, \\ \frac{DL}{D\mathbf{A}}(\mathbf{A}) &= \frac{1}{\|M_1 - M_2\|} \left\langle \frac{DM_1}{D\mathbf{A}} - \frac{DM_2}{D\mathbf{A}}, M_1 - M_2 \right\rangle. \end{aligned} \quad (3.20)$$

La formulation forte s’écrit :

$$\left\langle \Delta\mathbf{A}, \left(\frac{DM_1}{D\mathbf{A}} - \frac{DM_2}{D\mathbf{A}} \right)^T (M_1 - M_2) \right\rangle = -\|M_1 - M_2\|^2. \quad (3.21)$$

3.3.2 Contacts

Contact avec un demi-plan fixe

On impose aux solides du système de rester hors du demi-plan

$$D = \{x \in \mathbb{R}^2, \langle x, n \rangle \leq \lambda\}, \quad (3.22)$$

dans lequel le vecteur normal unitaire n et le scalaire λ sont donnés.

A chaque intrusion d’un solide dans D on introduit **un nouveau curseur** M correspondant au point de la frontière du solide ayant pénétré le plus profondément dans D c’est à dire minimisant $\langle M, n \rangle$ comme l’illustre la figure 28. La recherche de ce minimum se fait selon des techniques classiques (dichotomie/méthode de Newton quand la dérivée seconde M_{ss} n’est pas nulle).

Chaque curseur M introduit au cours d’un contact génère une équation fixant un “demi degré de liberté” sur l’évolution de \mathbf{A} :

$$\left\langle \Delta\mathbf{A}, \frac{DM^T}{D\mathbf{A}} n \right\rangle \geq -(\langle M, n \rangle - \lambda). \quad (3.23)$$

Cette inégalité est obtenue en écrivant les deux équations définissant le curseur M dépendant de \mathbf{A} et de son abscisse sur le contour du solide (on suppose en outre que la tangente au contour en M est définie) :

$$\begin{aligned} \langle M, n \rangle &= \lambda, \\ \langle M_s, n \rangle &= 0. \end{aligned} \quad (3.24)$$

On écrit ensuite ces conditions en $\mathbf{A} + \Delta\mathbf{A}$.

Lors d’un changement de la déformation le curseur doit être **mis à jour** pour continuer à minimiser $\langle M, n \rangle$.

Lorsque le curseur sort de D on le **supprime** de la liste des curseurs associés à la contrainte.

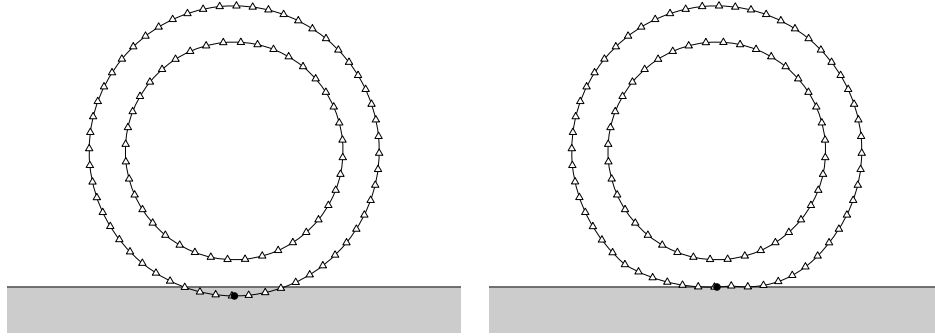


Fig. 28 - (a) Curseur généré par l'intrusion de l'anneau déformable dans le demi-plan grisé. Les triangles représentent les points d'échantillonnage des contours du solide. Le point noir correspond à l'unique curseur associé à la contrainte. (b) Etat du système après un déplacement dans la direction du "gradient de la contrainte". Cette direction correspond localement au plus court déplacement rendant la condition vérifiée.

Contact entre deux solides

La contrainte est la **non-intersection des intérieurs** des solides pris deux à deux.

De façon similaire au cas du contact avec un demi-plan (lorsque deux solides S_1 et S_2 s'intersectent (on peut avoir $S_1 = S_2$)) on introduit **deux nouveaux curseurs** M_1 et M_2 : un sur chaque solide. M_1 est sur la portion de surface de S_1 incluse dans S_2 et réciproquement. Les normales aux deux curseurs doivent être confondues.

Chaque paire de curseurs supprime un demi degré de liberté au système ; si n est un vecteur directeur unitaire de la normale commune aux deux curseurs on a :

$$\left\langle \frac{DM_1^T}{D\mathbf{A}} n - \frac{DM_2^T}{D\mathbf{A}} n, \Delta\mathbf{A} \right\rangle \geq \langle M_2 - M_1, n \rangle. \quad (3.25)$$

Comme dans le cas précédent les deux curseurs doivent être mis à jour lors de l'évolution de l'état pour que leurs normales restent confondues.

Lorsque M_1 sort du solide S_2 M_2 doit sortir en même temps de S_1 et on peut alors éliminer les deux curseurs.

3.3.3 Lois de comportement élastique

Les lois de comportement usuellement employées sont des lois linéarisées. Les équations sont alors simplifiées accélérant les traitements au détriment du réalisme physique.

Notre modèle est conçu indépendamment de toute hypothèse sur la complexité de la loi de comportement. Pour nos essais nous avons utilisé le modèle de matériau homogène de Saint-Venant–Kirchhoff. La densité d'énergie élastique de ces matériaux

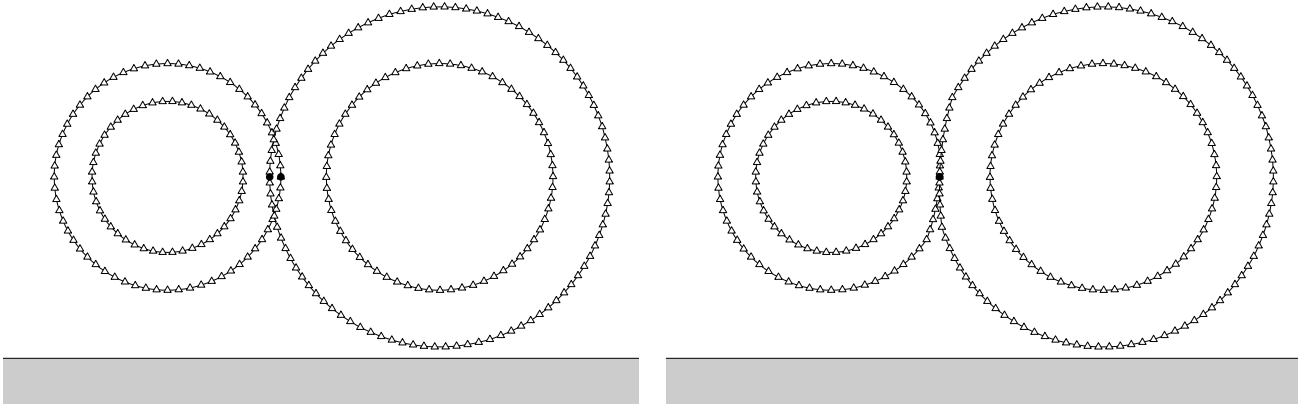


Fig. 29 - (a) Curseurs générés par l’interpénétration de deux anneaux déformables. Les triangles représentent les points d’échantillonnage du solide ; les points sont les deux curseurs. (b) Etat du système après un déplacement dans la direction du “gradient de la contrainte”.

s’écrit :

$$w(E) = \frac{\lambda}{2}(\text{Tr } E)^2 + \mu \text{Tr}(E^2), \quad (3.26)$$

où λ et μ sont les coefficients de Lamé définis dans le chapitre 3 de la partie *Outils*.

Comme le souligne Ciarlet dans [30] rien n’empêche dans ce modèle le déterminant du gradient de la déformation de devenir nul voire négatif. L’utilisation de lois de comportement faisant effectivement intervenir le déterminant permettrait de pallier à ce défaut (matériaux d’Odgen).

Jouve [69] propose des lois de comportement non isotropes modélisant les tissus de l’œil humain. L’utilisation de telles lois est naturelle pour la modélisation de tissus et augmenterait encore le réalisme physique du modèle.

3.4 Contrôle

Dans cette section consacrée à l’algorithme de contrôle nous détaillons l’algorithme de détection des contacts entre les solides puis la méthode itérative d’optimisation employée pour maintenir le système dans un minimum d’énergie tout en respectant les contraintes.

3.4.1 Détection des interpénétrations

Les interpénétrations entre solides se traduisent géométriquement par une intersection de leurs surfaces.

Les algorithmes existants de détection des intersections sont de deux types selon que la trajectoire des objets est connue ou non.

Dans le premier cas (trajectoire connue) seuls les objets rigides sont pris en compte dans la littérature. Cette restriction permet de propager de l’information

entre les différentes positions Γ et d'augmenter ainsi l'efficacité des algorithmes par rapport à ceux du second type Γ qui soit repartent "de zéro" à chaque nouvelle position Γ soit mettent à jour un "graphe de proximité" après chaque déplacement.

L'article récent de Lin et Manocha [77] décrit une solution dans le cas de trajectoires quelconques d'objets rigides ; on y trouvera aussi une introduction au problème et les références aux travaux antérieurs.

Dans notre cas Γ nous avons choisi de conserver l'approche développée en 1992 dans notre modèle des δ -snakes [4]. Il s'agit d'un algorithme du premier type Γ adapté aux surfaces déformables.

Nous représentons chaque contour par un ensemble de points de densité δ fixée : pour tout point m de la surface Γ il existe toujours au moins un point M de l'échantillon vérifiant $d(m, M) < \delta$.

Cette contrainte de densité est imposée dans la *configuration de référence* Γ afin de pouvoir conserver le même échantillon au cours de l'évolution du système. Nous supposons que les déformations des solides sont telles que les échantillons déformés soient de densité δ' . Nous avons choisi pour nos tests la valeur $\delta' = 2\delta$.

Le problème de détection des intersections se ramène alors à la recherche des paires de points distants d'au plus δ' dans la réunion des échantillons. Les paires de points vérifiant cette propriété sont communiquées à l'algorithme gérant les contraintes qui essaye d'en déduire deux curseurs aux tangentes parallèles (voir section précédente).

L'hypothèse sur les densités Γ ainsi que l'imposition d'un majorant sur la distance que peut parcourir un point de la surface d'un solide entre deux états successifs Γ permettent d'affirmer que cette méthode donne toutes les paires de curseurs.

Nous allons détailler l'algorithme utilisé dans [4] pour énumérer les paires de points.

La réunion des échantillons de tous les contours des solides sert à construire un *quadtree* Γ tel que celui représenté figure 30. Chaque feuille de l'arbre est soit vide Γ soit occupée par un seul point. Un sommet interne de l'arbre a exactement 4 fils.

L'ensemble des feuilles de l'arbre contenant un point est ensuite parcouru. Pour chaque point m Γ on utilise la structure locale de l'arbre pour énumérer les points se trouvant dans la *moitié droite* du carré de centre m et de côté $2\delta'$ (l'orientation des axes de ce carré est identique à celles du *quadtree*). Ne prendre qu'une moitié du carré permet d'éliminer la quasi-totalité des paires symétriques (seules les paires de points situés sur une même verticale seront traitées deux fois). On remonte tout d'abord dans l'arbre jusqu'à ce que le nœud courant contienne entièrement le demi-carré. Sur la figure 31 Γ le demi-carré figure en gris clair Γ et le rectangle correspondant à ce nœud en trait tireté fort. On explore ensuite tous les descendants de ce nœud en testant leur intersection avec le demi-carré. Sur la figure Γ les nœuds explorés sont en gris foncé.

Nous n'avons pas cherché à évaluer la complexité de cet algorithme d'énumération des paires de points. Il ne nécessite que des calculs de distances entre deux points Γ plus simples que les calculs d'intersections entre deux triangles Γ utilisés par Baraff et Witkin dans [8].

On peut aussi envisager l'utilisation de la triangulation de Delaunay des points Γ

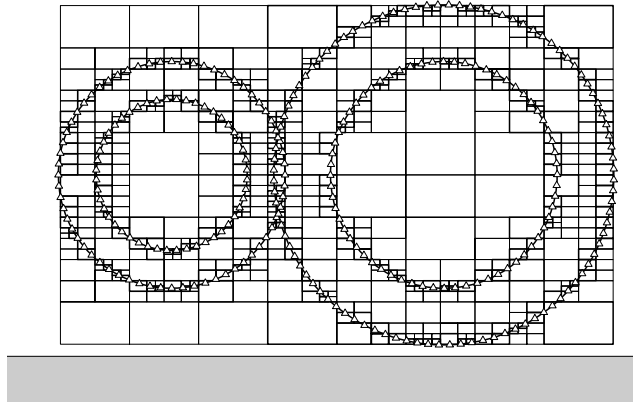


Fig. 30 - Quadtree formé à partir de quatre échantillons de points représentant les frontières de deux anneaux. Les bornes horizontale et verticale de l'arbre correspondent au plus petit rectangle englobant les échantillons de tous les solides. Les noeuds dessinés en trait fort sont les feuilles contenant un point d'échantillonnage.

qui est particulièrement adaptée à cette énumération.

Une paire de curseurs fournie par l'algorithme d'énumération est ensuite déplacée pour que les curseurs aient leurs normales confondues; on peut alors décider de garder ou non la paire de curseurs. Dans le cas où l'un des curseurs est sur un point anguleux de son contour on ne déplace que l'autre curseur.

Dans l'algorithme itératif d'optimisation utilisé pour calculer l'état d'énergie minimale (localement) on peut ne calculer les nouveaux curseurs que périodiquement en se contentant de mettre à jour les anciens à chaque itération. C'est ce que nous avons fait dans les exemples de la section suivante.

3.4.2 Optimisation sous contraintes

L'algorithme de contrôle adopté est itératif: partant d'un état \mathbf{A}_k une itération consiste d'abord à calculer les contraintes Γ et ensuite à déterminer un déplacement $\Delta \mathbf{A}$ du système (il s'agit d'un déplacement dans l'espace des états). On en déduit l'état suivant $\mathbf{A}_{k+1} = \mathbf{A}_k + \Delta \mathbf{A}$.

Le choix d'un algorithme itératif est dicté par l'existence de contraintes conditionnelles: ne connaissant pas globalement la géométrie de l'espace libre on ne peut pas optimiser globalement.

Les contraintes représentées par une "énergie" sont ajoutées à l'énergie potentielle du système Γ pondérées par des coefficients que l'on augmentera progressivement (méthode de pénalités voir [29]).

On note $G(\mathbf{A})$ le gradient de cette somme selon \mathbf{A} . Nous imposons à la norme du déplacement $\Delta \mathbf{A}$ d'être inférieure ou égale à un scalaire R . Cette condition est nécessaire pour que l'algorithme de détection des interpénétrations fonctionne comme nous l'avons vu plus haut.

On cherche alors un déplacement $\Delta \mathbf{A}$ solution du problème suivant (c est le

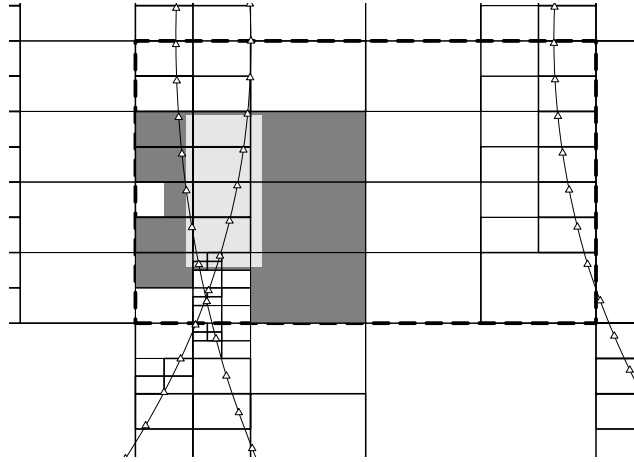


Fig. 31 - *Détail de l'arbre montrant le fonctionnement de l'algorithme pour un point m . Ce point est au milieu du côté gauche du demi-carré gris clair de largeur δ' et de hauteur $2\delta'$. Les nœuds intersectant ce demi-carré sont en gris foncé ; le nœud contenant m n'est pas concerné. Le rectangle tireté représente le plus petit nœud de l'arbre contenant entièrement le demi-carré.*

nombre de contraintes dans l'état \mathbf{A}) :

$$\left\{ \begin{array}{l} \langle \Delta \mathbf{A}, G(\mathbf{A}) \rangle \text{ minimal} \\ \langle \Delta \mathbf{A}, U_1 \rangle \geq l_1 \\ \langle \Delta \mathbf{A}, U_2 \rangle \geq l_2 \\ \dots \\ \langle \Delta \mathbf{A}, U_c \rangle \geq l_c \end{array} \right. \quad (3.27)$$

Il s'agit d'un problème de programmation linéaire. La littérature traite abondamment de ce sujet. Les ouvrages de Minoux [83] et Ciarlet [29] fournissent les notions et références de base dans ce domaine. Nous ajoutons ici la contrainte non linéaire $\|\Delta \mathbf{A}\| < R$.

Nous avons pour nos premiers tests préféré utiliser une méthode intuitive sans doute beaucoup moins efficace que les méthodes existantes adaptées au problème mais très facile à implémenter.

Une itération consiste dans notre cas à calculer $G(\mathbf{A})$ et les contraintes puis $H = \sum_c l_i U_i$. On ramène la norme de ces deux vecteurs à R si elle est supérieure puis on choisit $\Delta \mathbf{A} = 0.5G(\mathbf{A}) + H$.

Cette méthode a permis d'obtenir les résultats présentés dans la section suivante.

3.5 Résultats

Dans tous les exemples présentés on montre l'état d'énergie minimal atteint à partir d'un état du système donné.

En pratique l'algorithme de contrôle itère en permanence pour rapprocher le système de l'état d'équilibre le plus proche. Les perturbations extérieures du système

sont donc prises en compte immédiatement. L'optimisation se fait donc en principe en partant d'états "proches" de l'équilibre.

Les exemples donnés ici illustrent le fonctionnement du modèle et sortent un peu de ce cadre : les états initiaux sont "éloignés" de l'équilibre.

La convergence des premiers systèmes ne faisant intervenir que des solides rigides (3 degrés de liberté chacun) est de l'ordre de la seconde. Par contre la convergence des systèmes comprenant un ou plusieurs solides déformables (environ une centaine de degrés de liberté) est de l'ordre de l'heure.

Nous montrons tout d'abord comment le modèle traite le problème de l'équilibre d'un système de solides rigides puis élastiques. Ensuite nous voyons des systèmes dans lesquels cohabitent ces deux types de solides. Finalement des exemples de systèmes complexes permettent de se faire une idée des applications chirurgicales du modèle.

T rigide reposant sur un plan

Partant de deux états différents du même système les figures 32 et 33 montrent deux états d'équilibre possibles du système.



Fig. 32 - *T rigide droit au dessus d'un plan horizontal. (a) Etat initial. (b) Etat final.*

Dans les deux cas le contact est assuré par seulement deux curseurs situés aux points anguleux de l'objet. Dans le premier cas aux extrémités du segment d'appui et dans le second cas aux deux points de contact.



Fig. 33 - *T rigide incliné au dessus d'un plan horizontal. (a) Etat initial. (b) Etat final.*

Etoile rigide reposant sur un plan

Le cas d'objets présentant des pointes est traité correctement comme l'illustre la figure 34 dans le cas d'une étoile.

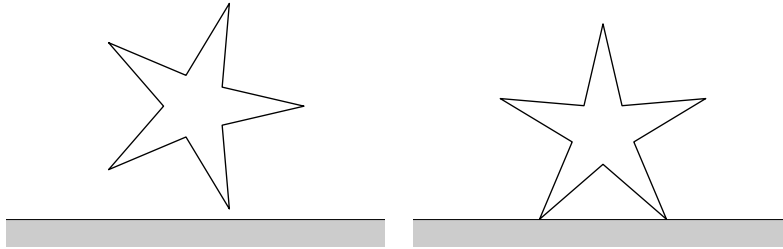


Fig. 34 - *Etoile rigide reposant sur un plan horizontal. (a) Etat initial. (b) Etat final.*

Deux T posés l'un sur l'autre

La figure 35 montre deux solides rigides T en contact avec un plan horizontal posés l'un sur l'autre. Cinq curseurs représentent les contacts. Le cas des points anguleux nécessite un traitement spécial lors de la recherche des paires de curseurs intervenant dans un contact.

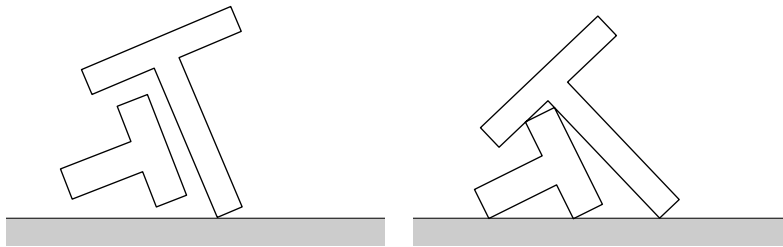


Fig. 35 - *Deux T rigides posés l'un sur l'autre sur un plan horizontal. (a) Etat initial. (b) Etat final.*

Deux T repoussés entre deux parois

Nous avons repris le système du paragraphe précédent en rajoutant deux parois verticales. Sur la figure 36 la paroi de gauche est immobile et la paroi de droite se déplace vers la gauche.

Cet exemple illustre la liaison entre le système et le “temps”. L'algorithme de contrôle effectue en continu des itérations approchant le système d'un état d'équilibre. Certains objets ou contraintes de la scène sont modifiés en continu en fonction de données extérieures.

Ici le déplacement de la contrainte “paroi droite” est lié au nombre d'itérations de l'algorithme de contrôle.

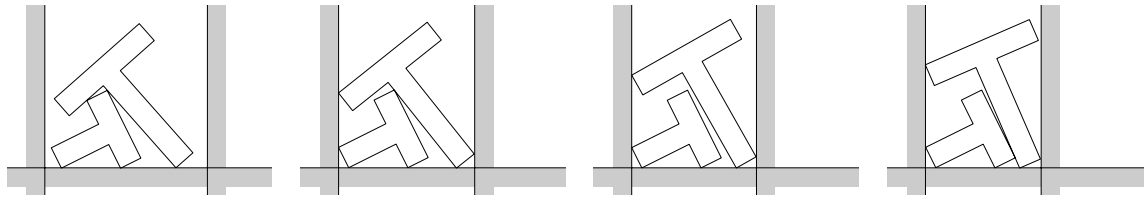


Fig. 36 - Deux T rigides posés l'un sur l'autre sur un plan horizontal, entre deux parois qui se rapprochent.

Anneau élastique reposant sur un plan

Sur la figure 37 on montre un anneau élastique reposant sur un plan horizontal. Les coefficients de la loi de comportement élastique sont les mêmes dans les deux cas ; seul change le nombre d'éléments représentant l'anneau. On voit que les éléments employés n'imposent pas la continuité du gradient de la déformation entre deux éléments contigus. On constate aussi la nécessité de mailler plus finement les parties des solides soumises à de fortes contraintes.

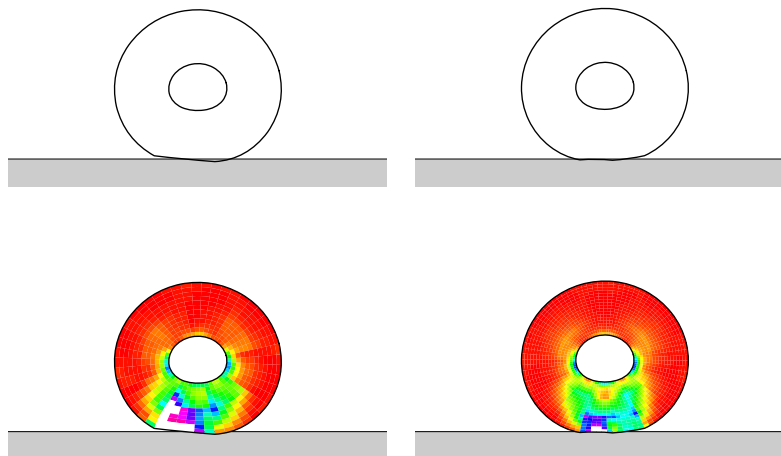


Fig. 37 - Anneau élastique reposant sur un plan horizontal. La déformation est due à l'action du champ de pesanteur, compensé par les contraintes internes dues à l'élasticité. En haut. (a) Anneau constitué de 8×2 éléments (circonférence \times épaisseur). (b) Anneau constitué de 15×3 éléments. En bas. Champs d'énergie potentielle élastique correspondants.

Objet élastique dans une cavité élastique

La figure 38 montre l'équilibre d'un solide élastique déposé à l'intérieur d'une cavité elle-même élastique.

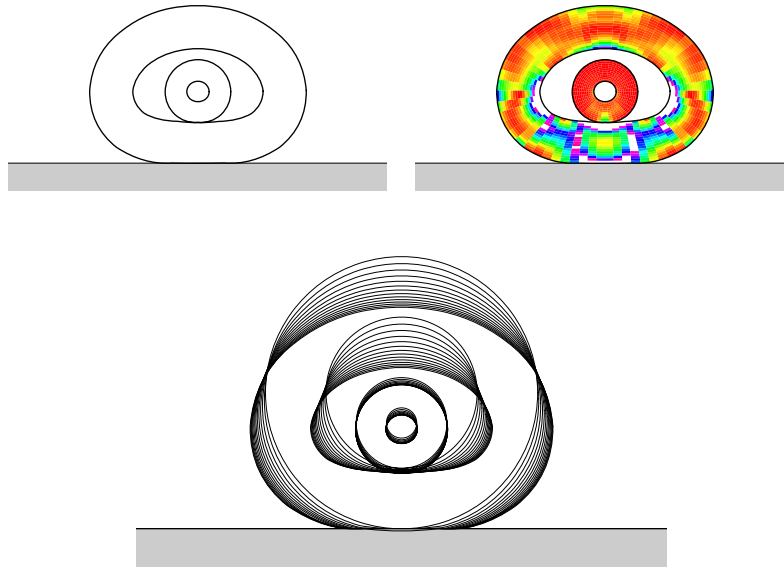


Fig. 38 - Anneau élastique reposant à l'intérieur d'un autre anneau également élastique. En haut. (a) Etat d'équilibre. (b) Energie potentielle élastique. En bas. Evolution de l'état du système pendant l'optimisation.

Contact entre un solide rigide et un solide élastique

Nous avons créé une cavité avec trois parois. Au fond de cette cavité un T rigide est posé sur un anneau élastique (figure 39).

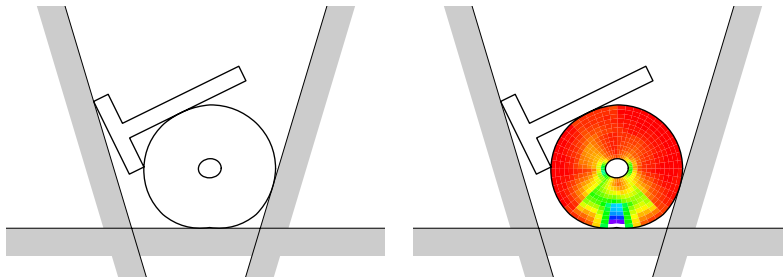


Fig. 39 - T rigide posé sur un anneau élastique au fond d'une cavité. (a) Etat d'équilibre. (b) Energie potentielle élastique dans l'anneau.

Plusieurs "organes" en contact dans une cavité

La figure 40 illustre un équilibre avec des contacts multiples entre des solides déformables.

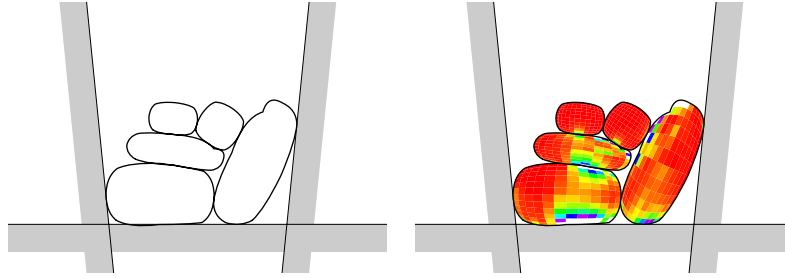


Fig. 40 - 5 solides élastiques dans une cavité. Les solides sont tous constitués de 2×2 éléments. Ce nombre est insuffisant, comme on peut le voir sur le solide le plus à droite. (a) Etat d'équilibre. (b) Energie potentielle élastique dans les solides.

3.6 Conclusion

Dans ce chapitre nous avons tout d'abord défini les propriétés requises par l'utilisation "chirurgicale" d'un modèle de système de solides.

Ensuite nous avons présenté en partant de l'hypothèse d'évolution quasi statique du système un modèle permettant de satisfaire ces exigences.

Ce modèle est formé de trois modules indépendants : la représentation de la géométrie et des déformations des solides, le calcul et la mise à jour des contraintes, l'algorithme de contrôle de l'évolution système.

Nous avons défini les fonctionnalités remplies par chacun de ces modules.

Pour le premier module, géométrie et déformations, nous avons montré comment implémenter ces fonctionnalités dans deux cas :

- les solides déformables représentés par un maillage de type éléments finis et régis par une loi de comportement hyperélastique quelconque
- les solides rigides polygonaux.

Dans le module gérant les contraintes, nous avons traduit en équations sur le vecteur d'état du système deux contraintes géométriques fixes et deux contraintes conditionnelles (contacts).

Le module de contrôle a pour première tâche la mise à jour de l'ensemble des contraintes, en détectant les intersections et en calculant les équations correspondantes. Sa seconde tâche consiste à modifier l'état du système itérativement pour se rapprocher d'un état d'équilibre en vérifiant les contraintes à chaque instant.

Nous avons vu sur quelques exemples que le modèle présenté ici permet d'obtenir des résultats réalistes et conformes à l'intuition physique.

L'hypothèse d'évolution quasi-statique permet de ne pas tenir compte de l'inertie. Le problème des oscillations évoqué dans le chapitre 2 n'existe pas dans ce cas. La formulation des contraintes de contact s'en trouve également simplifiée. L'algorithme de contrôle est ainsi beaucoup plus robuste qu'en présence de forces de réaction, dont on ne maîtrise pas toujours l'apport énergétique. L'évolution quasi-statique est

justifiée par l'absence de chocs par la vitesse réduite des déplacements et par les masses proches des différents solides en contact.

La conception modulaire rend possible la coexistence dans le système de solides représentés de façon quelconque ayant des comportements différents (solides rigides, mous...). Elle se prête également très bien à la programmation dans un langage orienté objets (nous avons utilisé intensivement le langage **C++**).

L'utilisation de modèles de type éléments finis permet d'atteindre un réalisme physique satisfaisant. En particulier on peut estimer l'erreur faite dans le calcul de l'énergie et provoquer le remaillage local des solides.

Ce maillage adaptatif est l'une des prochaines étapes de nos recherches. L'autre étape essentielle est le passage à la dimension 3. Ce passage s'accompagne d'une augmentation sensible du nombre de degrés de libertés du système et de la taille des échantillons des surfaces des solides.

Le modèle présenté possède encore quelques défauts :

- Les éléments utilisés n'empêchent pas l'apparition de discontinuités sur les contours comme nous l'avons vu dans l'exemple de la figure 37. Une solution serait de mémoriser en chaque noeud le gradient de la déformation par exemple en utilisant des éléments à 16 noeuds dont les positions sont reliées par des contraintes linéaires assurant la continuité du gradient.
- Si l'utilisation d'éléments finis permet d'obtenir des déformations réalistes de grande amplitude la dimension du vecteur d'état croît très rapidement avec le nombre d'éléments. Pour les solides "moyennement rigides" des **déformations globales** paramétrées par une vingtaine de coefficients donneraient le même résultat avec un temps de calcul beaucoup plus faible.
- La lenteur de l'algorithme de contrôle ne permet une vitesse de calcul compatible avec une interaction que dans le cas de systèmes constitués de solides rigides. Cet algorithme doit donc être amélioré. L'utilisation de la dérivée seconde de l'énergie en fonction de **A** permettrait d'utiliser des méthodes plus rapides.
- L'algorithme d'optimisation adopté ne permet pas d'obtenir un critère de convergence commun à tous les systèmes.

Les perspectives d'utilisation et d'extension des travaux présentés dans ce chapitre sont discutées dans la conclusion de cette partie.

Chapitre 4

Conclusion

Dans cette partie nous avons abordé le problème de l'animation de systèmes de solides rigides et déformables contrôlés par les lois de la physique.

Tout d'abord dans le chapitre 2 nous avons présenté les travaux existants en dissociant chaque modèle selon le schéma : géométrie et déformations / contraintes / contrôle.

Ensuite dans le chapitre 3 après avoir défini les propriétés que doit vérifier un modèle pour permettre son utilisation "chirurgicale" nous avons présenté les résultats de nos recherches sur le sujet.

Ces travaux partent de l'hypothèse d'une évolution quasi statique du système. Elle permet de simplifier les équations et de rendre l'algorithme de contrôle du système plus stable que dans les modèles usuels faisant intervenir l'inertie des solides et un contrôle par forces.

D'autre part l'utilisation de lois de comportement issues de la mécanique des milieux continus et la représentation des solides par éléments finis permet d'atteindre un réalisme satisfaisant.

Les résultats obtenus en dimension 2 montrent que l'on peut modéliser des systèmes d'une complexité permettant de construire un prototype de simulateur chirurgical.

Après être passé à la dimension 3 et avoir introduit le remaillage adaptatif des éléments la simulation de gestes tels que la rétroperitonéoscopie sera envisageable.

Toutefois l'acquisition des surfaces et des constantes intervenant dans les lois de comportement des organes reste toujours un problème. Pour l'acquisition des surfaces des atlas anatomiques "électroniques" du corps humain sont en constitution ; il faut néanmoins tenir compte des variations anatomiques d'un individu à l'autre par exemple en mettant en correspondance un ou plusieurs atlas anatomiques avec les coupes du patient. Nous avons suggéré deux méthodes pour évaluer les coefficients paramétrant le comportement des organes : placer des capteurs sur les instruments du chirurgien et utiliser un expert pour ajuster les paramètres d'un simulateur.

Une fois le modèle tridimensionnel opérationnel on pourra aussi essayer de coupler un solide du système à un manipulateur à retour d'effort.

Troisième partie

Outils

Chapitre 1

Introduction

La réalisation du système d'assistance à la rétroperitonéoscopie présenté dans la première partie soulève quelques problèmes : la **localisation précise** de billes de plomb dans une image "scanner" le **calibrage** des marqueurs infra-rouge le **recalage** du support de calibrage du médiastinoscope la **mise en correspondance** des centres des billes mesurés sur le patient avec les mêmes centres localisés dans les coupes scanner.

De même l'étude des modèles d'animation et de modelage par ordinateur réalisée dans la deuxième partie nécessite la connaissance des équations de la **mécanique des milieux continus** des lois de **comportement élastique** des solides des méthodes d'**éléments finis** des algorithmes d'**optimisation** et des méthodes d'**intégration** des équations différentielles. Nous avons aussi abordé certains problèmes de géométrie algorithmique comme l'énumération des couples de points dont la distance est inférieure à un certain seuil dans un nuage de points.

Nous avons choisi de présenter dans cette partie quatre de ces problèmes.

Le chapitre 2 traite des différentes représentations des rotations. Représenter les rotations et connaître leurs propriétés est important dans les problèmes de recalage où l'inconnue est une transformation rigide dont la partie linéaire est une rotation.

Après avoir introduit les définitions essentielles nous voyons comment sont représentées les rotations en dimensions 2 et 3 et comment passer d'une représentation à l'autre de façon numériquement fiable. Sur ce point nous proposons quelques améliorations aux formules existantes instables dans certains cas. Ce chapitre se termine par une étude de la structure de l'ensemble SO_4 des rotations en dimension 4 et par une étude des éléments caractéristiques de ses éléments (sous-espaces globalement invariants nature de la restriction à ces sous-espaces).

Dans le chapitre 3 nous exposons l'essentiel des équations de la mécanique des milieux continus et des lois de comportement des solides élastiques qui nous ont servi dans la partie *Animation de systèmes de solides*. Ce chapitre ne contient aucune contribution originale. Nous avons toutefois établi un lien entre les différents tenseurs représentant les déformations et les contraintes au sein du solide et les parties rigide et non-rigide du gradient de la déformation.

L'étude du problème de la mise en correspondance de nuages de points appariés

est l'objet du chapitre 4. Les solutions présentées dans la littérature diffèrent en fonction de la représentation des rotations adoptée pour chercher la partie linéaire de la transformation inconnue. Nous présentons et comparons toutes les solutions existantes et proposons une nouvelle approche utilisant la représentation par vecteur rotation.

Enfin dans le chapitre 5 nous traitons trois aspects d'un problème de segmentation avec connaissance *a priori*. Il s'agit de rechercher dans une image bi- ou tri-dimensionnelle des objets circulaires ou elliptiques (sphériques ou "ellipsoïdaux" en dimension 3) d'une taille connue approximativement.

Tout d'abord nous tentons de trouver au pixel près tous les objets de ce type dans l'image. Ensuite connaissant la position d'un objet circulaire ou elliptique au pixel près on voit comment on peut optimiser la détection avec une résolution inférieure au pixel. Finalement nous étendons le problème à la détection globale d'une grille régulière d'objets circulaires. La connaissance précise de la projection de la grille peut ensuite être utilisée pour calibrer le système d'acquisition des images avec une très bonne précision.

Chapitre 2

Représentation des rotations

Ce chapitre présente les rotations et leurs différentes représentations. Nous donnons aussi des relations permettant de passer d'une représentation à l'autre.

Le choix d'une représentation pour les rotations intervient naturellement dans tous les problèmes dans lesquels l'inconnue est une rotation. C'est le cas par exemple dans le problème de mise en correspondance rigide de nuages de points appariés que nous étudions dans le chapitre 4 de cette partie.

Les différentes représentations des rotations interviennent également en animation par ordinateur pour représenter la position des solides rigides mais aussi pour interpoler les déplacements des objets entre deux positions fixées par l'opérateur.

Nous donnons tout d'abord dans la section 2.1 les définitions des objets utilisés dans ce chapitre.

Ensuite nous énonçons quelques résultats valables en dimension quelconque (section 2.2) avant de voir les représentations des rotations en dimensions 2 et 3 (sections 2.3 et 2.4) puis des algorithmes permettant de passer d'une représentation à l'autre en dimension 3 (section 2.5).

Enfin avant de conclure nous verrons dans la section 2.6 la structure du groupe SO_4 des rotations de \mathbb{R}^4 qui présente des caractéristiques que l'on ne retrouve pas dans les groupes SO_n pour $n \geq 5$. En particulier il existe un lien que nous expliciterons entre le groupe des quaternions unitaires et les deux sous-groupes distingués non triviaux de SO_4 . Nous verrons aussi quelles sont les caractéristiques géométriques des différents types de rotations en dimension 4.

2.1 Définitions

Cette section s'inspire des cours de mathématiques supérieures et spéciales de Delmas [34] et Moisan [86] ainsi que du livre de Mneimné et Testard [85].

Soit n un entier positif. On note M_n l'espace vectoriel des endomorphismes de \mathbb{R}^n et GL_n le groupe des endomorphismes inversibles de \mathbb{R}^n .

2.1.1 Rotations

Definition 8 (Endomorphisme orthogonal)

On qualifie d'orthogonal un endomorphisme f de \mathbb{R}^n qui conserve la norme :

$$\forall x \in \mathbb{R}^n, |f(x)| = |x|. \quad (2.1)$$

Ceci implique en particulier que f soit injectif et donc inversible : si $f(x) = f(y)$ alors $|f(x - y)| = 0$ et donc $x = y$. On note O_n l'ensemble des endomorphismes orthogonaux de \mathbb{R}^n . O_n est un sous-groupe (non commutatif) de GL_n .

On a les définitions équivalentes suivantes :

$$\begin{aligned} f &\in O_n, \\ \forall x \in \mathbb{R}^n, |f(x)| &= |x|, \\ \forall x, y \in \mathbb{R}^n, \langle f(x), f(y) \rangle &= \langle x, y \rangle, \\ M.M^T &= I, \end{aligned} \quad (2.2)$$

dans lesquelles M désigne la matrice de f dans une base orthonormée quelconque et $\langle x, y \rangle$ le produit scalaire usuel.

O_n est compact dans M_n et est constitué de deux composantes connexes correspondant aux endomorphismes de déterminants $+1$ et -1 .

Definition 9 (Groupe spécial orthogonal)

On appelle groupe spécial orthogonal noté SO_n le sous-groupe de O_n des endomorphismes orthogonaux de déterminant $+1$.

Definition 10 (Rotation)

On appelle rotation un élément de SO_n .

2.1.2 Définitions annexes

Les définitions suivantes seront utilisées dans la suite de ce chapitre.

Definition 11 (Réflexion)

La symétrie orthogonale par rapport à l'hyperplan H est appelée la réflexion d'hyperplan H . Par définition si r est la réflexion d'hyperplan H sa restriction à H est Id_H et sa restriction à la droite orthogonale à H notée H^\perp est $-Id_{H^\perp}$.

Definition 12 (Endomorphisme antisymétrique)

Un endomorphisme u est dit antisymétrique s'il vérifie $u + u^T = 0$. Nous noterons A_n l'ensemble de ces endomorphismes.

Definition 13 (Quaternions)

Le corps des quaternions (W.R. Hamilton 1843) noté \mathbb{H} est l'ensemble des couples $(s, u) \in \mathbb{R} \times \mathbb{R}^3$ muni d'une addition notée $+$ et d'une multiplication non commutative notée $*$ définies par (voir par exemple [47]) :

$$\begin{aligned} (s, u) + (t, v) &= (s + t, u + v), \\ (s, u) * (t, v) &= (st - \langle u, v \rangle, u \wedge v + sv + tu). \end{aligned} \quad (2.3)$$

Le quaternion $(0, 0)$ est l'élément neutre de l'addition et $(1, 0)$ celui de la multiplication. On identifie le réel x au quaternion $(x, 0)$ ainsi que le vecteur $u \in \mathbb{R}^3$ au quaternion $(0, u)$. \mathbb{H} est une \mathbb{R} -algèbre de dimension (réelle) 4.

Définition 14 (Conjugué d'un quaternion)

On définit le conjugué du quaternion $q = (s, u)$ par $\bar{q} = (s, -u)$. La conjugaison a les propriétés suivantes vis-à-vis des deux opérations :

$$\begin{aligned}\overline{q_1 + q_2} &= \bar{q}_1 + \bar{q}_2, \\ \overline{q_1 * q_2} &= \bar{q}_2 * \bar{q}_1.\end{aligned}\tag{2.4}$$

Définition 15 (Module d'un quaternion)

Le module $|q|$ d'un quaternion $q = (s, u)$ est le réel positif ou nul défini par $|q|^2 = q * \bar{q} = s^2 + |u|^2$. On a :

$$\begin{aligned}|q_1 * q_2| &= |q_1| |q_2|, \\ |\bar{q}| &= |q|.\end{aligned}\tag{2.5}$$

Définition 16 (Inverse d'un quaternion)

L'inverse du quaternion $q \in \mathbb{H} - \{0\}$ est le quaternion $q^{-1} = \frac{\bar{q}}{|q|^2}$.

Définition 17 (Produit scalaire de deux quaternions)

On définit le produit scalaire $(s, u).(t, v)$ des deux quaternions (s, u) et (t, v) par $(s, u).(t, v) = st + \langle u, v \rangle$. On peut remarquer que $q_1.q_2$ est la partie réelle du produit $q_1 * \bar{q}_2$:

$$q_1.q_2 = \frac{q_1 * \bar{q}_2 + q_2 * \bar{q}_1}{2}.\tag{2.6}$$

Le produit scalaire des quaternions est compatible avec le produit scalaire dans \mathbb{R}^3 : pour $u, v \in \mathbb{R}^3$ on a bien $u.v = \langle u, v \rangle$.

\mathbb{H} peut aussi être vu comme une extension du corps \mathbb{R} (voir par exemple [64]) en définissant les trois nombres "imaginaires" i, j, k vérifiant :

$$\begin{aligned}i^2 &= -1, \quad j^2 = -1, \quad k^2 = -1, \\ ij &= k, \quad jk = i, \quad ki = j, \\ ji &= -k, \quad kj = -i, \quad ik = -j.\end{aligned}\tag{2.7}$$

Un quaternion s'écrit alors comme combinaison linéaire à coefficients réels des vecteurs de la base $(1, i, j, k)$.

Ces deux représentations des quaternions sont équivalentes et liées par la relation $(s, u) = s + u_1i + u_2j + u_3k$.

Définition 18 (Quaternion unitaire)

On qualifie d'unitaire un quaternion de module 1. Nous noterons \mathbb{H}_1 l'ensemble des quaternions unitaires. $(\mathbb{H}_1, *)$ est un groupe non commutatif.

Definition 19 (Complexe unitaire)

Nous noterons \mathbb{C}_1 l'ensemble des complexes de module 1 appelés complexes unitaires.

Definition 20 (Exponentielle)

On appelle l'exponentielle de l'endomorphisme u la somme de la série normalement convergente

$$\exp u = \sum_{i \in \mathbb{N}} \frac{u^i}{i!}. \quad (2.8)$$

$\exp u$ est dans GL_n . On a les propriétés suivantes :

$$\begin{aligned} \det(\exp u) &= e^{\text{Tr } u}, \\ (\exp u)^T &= \exp u^T, \\ \forall u, v \in M_n, uv = vu &\Rightarrow \exp(u + v) = (\exp u)(\exp v), \\ \forall u \in M_n, \forall p \in GL_n, &\exp(p^{-1}up) = p^{-1}(\exp u)p. \end{aligned} \quad (2.9)$$

Definition 21 (Sphère unité)

L'ensemble des vecteurs de \mathbb{R}^n de norme euclidienne 1 est appelé la sphère unité de \mathbb{R}^n et est noté S_{n-1} . Cet ensemble est une sous-variété de \mathbb{R}^n de dimension $n - 1$.

Definition 22 (Boule ouverte)

On appelle boule ouverte de \mathbb{R}^n de centre $a \in \mathbb{R}^n$ et de rayon $r \geq 0$ l'ensemble

$$B_n(a, r) = \{u \in \mathbb{R}^n, |u - a| < r\}. \quad (2.10)$$

2.2 Propriétés

Avant de voir en détail les représentations possibles des rotations en dimension 2 et 3 dans les sections suivantes nous pouvons énoncer quelques résultats valables en dimension n quelconque.

2.2.1 Sous-espaces globalement invariants

Soit $f \in O_n$ il existe une base orthonormée de \mathbb{R}^n dans laquelle la matrice de f est de la forme diagonale par blocs :

$$\text{Diag}(I_p, -I_q, R_{\theta_1}, \dots, R_{\theta_r}), \quad (2.11)$$

dans laquelle I_k est la matrice identité de dimension k et R_θ celle de la rotation "plane" à deux dimensions d'angle $\theta \neq 0 \pmod{2\pi}$ (le cas $\theta = 0 \pmod{2\pi}$ correspond à I_2 et se retrouve donc dans la partie I_p et le cas $\theta = \pi \pmod{2\pi}$ se retrouve de même dans la partie $-I_q$). R_θ s'écrit :

$$R_\theta = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}. \quad (2.12)$$

Les sous-espaces vectoriels globalement invariants par f de dimension minimale sont donc de dimension 1 ou 2. Il s'agit des droites du sous-espace de dimension p sur lequel la restriction de f est $Id_{\mathbb{R}^p}$ et des droites du sous-espace de dimension q sur lequel la restriction de f est $-Id_{\mathbb{R}^q}$ et des r plans de dimension 2 sur lesquels la restriction de f est une rotation plane (de \mathbb{R}^2) d'angle θ_k .

Le déterminant de f est alors le produit des déterminants des blocs Γ c'est à dire $(-1)^q$. De plus on doit avoir $p + q + 2r = n$.

De fait dans le cas où f est une rotation en dimension 2 on peut rencontrer les valeurs suivantes du triplet (p, q, r) : $(2, 0, 0)$ (identité) $\Gamma(0, 2, 0)$ (symétrie centrale) Γ et $(0, 0, 1)$ (rotation non triviale). En dimension 3 toujours pour une rotation on a les possibilités suivantes: $(3, 0, 0)$ (identité) $\Gamma(1, 2, 0)$ (demi-tour Γ appelé aussi retournement) $\Gamma(1, 0, 1)$ (rotation non triviale).

2.2.2 Produit de réflexions

Tout endomorphisme orthogonal $f \in O_n$ s'écrit comme produit de $k \leq n$ réflexions [86]. Le déterminant de f est alors $(-1)^k$.

Dans le cas d'une rotation k doit donc être pair; en particulier pour $n = 2$ ou $n = 3$ k ne peut prendre que les valeurs 0 (identité) et 2. Pour $n = 4$ on a de même $k \in \{0, 2, 4\}$.

2.2.3 Exponentielle d'un endomorphisme antisymétrique

On établit que l'exponentielle d'un élément de A_n est dans SO_n (conséquence de 2.9) Γ et réciproquement Γ que pour toute rotation $u \in SO_n$ il existe un endomorphisme antisymétrique $h \in A_n$ tel que $u = \exp h$ (voir [85]).

SO_n est une sous-variété de M_n ; l'espace vectoriel tangent à SO_n en Id est A_n Γ de dimension $\frac{n(n-1)}{2}$.

SO_2 est donc une sous-variété de M_2 de dimension 1 Γ et SO_3 une sous-variété de M_3 de dimension 3.

On peut calculer numériquement l'exponentielle d'une matrice antisymétrique H en diagonalisant la matrice symétrique positive $-H^2$ dans une base orthonormée :

$$-H^2 = \Omega \operatorname{Diag}_{i=1\dots n} (\lambda_i^2) \Omega^T. \quad (2.13)$$

Dans cette dernière expression Ω est la matrice orthogonale de la base de diagonalisation ($\Omega \Omega^T = I_n$) Γ et les λ_i sont les racines carrées des valeurs propres de $-H^2$ qui sont toutes positives. On a alors Γ en regroupant les termes pairs et impairs de la série définissant $\exp H$ et en reconnaissant les développements des fonctions sin et cos :

$$\exp H = \Omega \operatorname{Diag}_{i=1\dots n} (\cos \lambda_i) \Omega^T + H \Omega \operatorname{Diag}_{i=1\dots n} \left(\frac{\sin \lambda_i}{\lambda_i} \right) \Omega^T. \quad (2.14)$$

La fonction $\lambda \mapsto \frac{\sin(\lambda)}{\lambda}$ est prolongeable en 0 Γ et de classe C^∞ .

2.2.4 Matrice orthogonale

Un endomorphisme dont la matrice M dans la base canonique de \mathbb{R}^n vérifie les deux équations :

$$\begin{aligned} MM^T &= I_n, \\ \det M &= 1, \end{aligned} \tag{2.15}$$

est une rotation. On peut donc représenter une rotation par sa matrice dans la base canonique de \mathbb{R}^n . Cette représentation nécessite n^2 scalaires vérifiant $\frac{n(n+1)}{2}$ contraintes non linéaires.

2.3 Représentations en dimension 2

2.3.1 Angle

Les rotations sont caractérisées par un scalaire θ : l'angle de la rotation. Notons $r_1(\theta) : \mathbb{R}/2\pi\mathbb{Z} \rightarrow SO_2$ la rotation d'angle θ . $r_1(\theta)$ est définie par sa matrice dans une base orthonormée directe :

$$\begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}. \tag{2.16}$$

On a les propriétés suivantes :

$$\begin{aligned} r_1(0) &= Id, \\ r_1(\theta)^{-1} &= r_1(-\theta), \\ r_1(\theta_1) \circ r_1(\theta_2) &= r_1(\theta_1 + \theta_2). \end{aligned} \tag{2.17}$$

Les groupes (SO_2, \circ) et $(\mathbb{R}/2\pi\mathbb{Z}, +)$ sont isomorphes ; un isomorphisme étant r_1 .

2.3.2 Matrice antisymétrique

Comme nous l'avons rappelé au paragraphe 2.2.3 l'exponentielle d'un endomorphisme antisymétrique est une rotation.

On a la propriété :

$$\forall h \in \mathbb{R}, \exp \tilde{h} = r_1(h), \tag{2.18}$$

en notant \tilde{h} l'endomorphisme antisymétrique de matrice

$$\begin{pmatrix} 0 & -h \\ h & 0 \end{pmatrix}. \tag{2.19}$$

L'angle de la rotation correspond donc à son "logarithme".

2.3.3 Complexe unitaire

On peut représenter une rotation de SO_2 par un nombre complexe de module 1. Notons $r_2(z) : \mathbb{C}_1 \rightarrow SO_2$ l'application définie par :

$$\forall u \in \mathbb{R}^2, r_2(z)u = zu. \quad (2.20)$$

On a les propriétés suivantes :

$$\begin{aligned} r_2(1) &= Id, \\ r_2(z)^{-1} &= r_2(\bar{z}), \\ r_2(z_1) \circ r_2(z_2) &= r_2(z_1 z_2). \end{aligned} \quad (2.21)$$

Les groupes (SO_2, \circ) et (\mathbb{C}_1, \cdot) sont isomorphes ; cet isomorphisme étant r_2 .

2.4 Représentations en dimension 3

Nous avons vu qu'en dimension 3 une rotation $r \in SO_3$ différente de l'identité possédait une droite invariante (sur laquelle sa restriction est l'identité). Cette droite est appelée l'axe de la rotation. La restriction de r au plan orthogonal à l'axe est une rotation $r' \in SO_2$. On appelle angle de r l'angle de r' . Le signe de l'angle et l'orientation de l'axe sont liés. L'identité peut être vue comme une rotation d'axe quelconque et d'angle nul.

Les représentations des rotations de SO_3 sont assez proches de celles des rotations de SO_2 . L'analogie tridimensionnelle de l'angle d'une rotation plane sera un vecteur de \mathbb{R}^3 appelé vecteur rotation et possédant les mêmes propriétés vis-à-vis de l'exponentielle. Les quaternions unitaires fournissent l'équivalent tridimensionnel des complexes unitaires.

2.4.1 Axe et angle

D'après ce qui précède on peut donc associer une rotation à un axe et un angle. Soit $r_1(k, \theta) : S_2 \times \mathbb{R}/2\pi\mathbb{Z} \rightarrow SO_3$ l'application définie par (formule de Rodrigues) :

$$\forall u \in \mathbb{R}^3, r_1(k, \theta)u = \langle u, k \rangle k + \cos\theta(u - \langle u, k \rangle k) + \sin\theta(k \wedge u). \quad (2.22)$$

Cette représentation possède les propriétés suivantes :

$$\begin{aligned} \forall k \in S_2, r_1(k, 0) &= Id, \\ r_1(k, \theta)^{-1} &= r_1(k, -\theta), \\ r_1(k, \theta_1) \circ r_1(k, \theta_2) &= r_1(k, \theta_1 + \theta_2), \\ r_1(k, \theta) &= r_1(-k, -\theta). \end{aligned} \quad (2.23)$$

Elle présente l'avantage d'avoir une interprétation géométrique immédiate mais plusieurs inconvénients : la multiplicité des représentations de l'identité l'absence de règle générale simple de composition et d'inversion et enfin l'ambiguïté sur le choix de l'orientation de l'axe.

2.4.2 Vecteur rotation

On peut lever l'ambiguïté sur l'orientation de l'axe en remarquant que r_1 ne dépend en fait que du produit $\omega = \theta k$. Définissons donc $r_2(\omega) : \overline{B_3(O, \pi)} \rightarrow SO_3$ par :

$$\forall u \in \mathbb{R}^3, r_2(\omega)u = \frac{\langle u, \omega \rangle}{|\omega|^2} \omega + \cos|\omega| \left(u - \frac{\langle u, \omega \rangle}{|\omega|^2} \omega \right) + \frac{\sin|\omega|}{|\omega|} (\omega \wedge u). \quad (2.24)$$

ω est appelé le vecteur rotation. On a $r_1(k, \theta) = r_2(\theta k)$ et les propriétés de r_2 se déduisent donc de celles de r_1 .

L'identité correspond maintenant au seul vecteur nul (quand ω est nul r_2 reste définie et est l'identité) et l'ambiguïté sur l'orientation de l'axe a disparu. Il subsiste néanmoins deux représentations possibles pour chaque demi-tour : ces derniers correspondent à $|\omega| = \pi$ et dans ce cas $r_2(\omega) = r_2(-\omega)$. De plus on n'a toujours pas de règle simple pour l'inversion et la composition.

2.4.3 Matrice antisymétrique

Comme nous l'avons vu en 2.2.3 toute rotation s'écrit comme l'exponentielle d'un endomorphisme antisymétrique. A_3 est un sous-espace vectoriel de M_3 de dimension 3. On peut donc associer à un vecteur $h \in \mathbb{R}^3$ un endomorphisme antisymétrique \tilde{h} défini par sa matrice dans la base canonique de \mathbb{R}^3 :

$$\begin{pmatrix} 0 & -h_3 & h_2 \\ h_3 & 0 & -h_1 \\ -h_2 & h_1 & 0 \end{pmatrix}. \quad (2.25)$$

Par un raisonnement similaire à celui que nous détaillons dans la section consacrée aux rotations en dimension 4 (voir 2.57) on obtient la propriété suivante :

$$\forall h \in \overline{B_3(O, \pi)}, \exp \tilde{h} = r_2(h). \quad (2.26)$$

Le vecteur rotation et le "logarithme" de la rotation sont donc intimement liés comme nous l'avions déjà observé en dimension 2.

2.4.4 Quaternion unitaire

De la même façon qu'une rotation en dimension 2 peut être représentée par un complexe unitaire une rotation en dimension 3 peut l'être par un quaternion unitaire. Définissons $r_3(q) : \mathbb{H}_1 \rightarrow SO_3$ par :

$$\forall u \in \mathbb{R}^3, r_3(q)u = q * u * \bar{q}. \quad (2.27)$$

$r_3(q)u$ est dans \mathbb{R}^3 car en remarquant que les quaternions de la forme $m = (0, u)$ sont caractérisés par $m = -\bar{m}$ on a pour tout quaternion unitaire q et tout vecteur $u \in \mathbb{R}^3$:

$$\begin{aligned} \overline{r_3(q)u} &= \overline{q * u * \bar{q}}, \\ &= q * \bar{u} * \bar{q}, \\ &= q * (-u) * \bar{q}, \\ &= -r_3(q)u. \end{aligned} \quad (2.28)$$

Pour montrer que $r_3(q)$ est effectivement une rotation Γ nous allons utiliser la propriété de conservation du produit scalaire 2.2 et la définition 2.6 Γ pour un quaternion unitaire $q \Gamma$ et deux vecteurs $u, v \in \mathbb{R}^3 \Gamma$ on a :

$$\begin{aligned}
\langle r_3(q)u, r_3(q)v \rangle &= \frac{(r_3(q)u) * (\overline{r_3(q)v}) + (r_3(q)v) * (\overline{r_3(q)u})}{2}, \\
&= \frac{(q * u * \bar{q}) * (q * \bar{v} * \bar{q}) + (q * v * \bar{q}) * (q * \bar{u} * \bar{q})}{2}, \\
&= q * \frac{u * \bar{v} + v * \bar{u}}{2} * \bar{q}, \\
&= q * \langle u, v \rangle * \bar{q}, \\
&= \langle u, v \rangle.
\end{aligned} \tag{2.29}$$

On a les propriétés suivantes :

$$\begin{aligned}
r_3(1) &= Id, \\
r_3(q)^{-1} &= r_3(\bar{q}), \\
r_3(q_1) \circ r_3(q_2) &= r_3(q_1 * q_2), \\
r_3(q) &= r_3(-q).
\end{aligned} \tag{2.30}$$

En particulier Γ la composée de deux rotations est représentée simplement par le produit de leurs quaternions. On conserve toujours la même indétermination sur l'orientation des axes. En considérant le quotient de \mathbb{H}_1 par la relation \mathcal{R} définie sur $\mathbb{H}_1 \times \mathbb{H}_1$ par $q_1 \mathcal{R} q_2 \equiv (q_1 = q_2) \vee (q_1 = -q_2) \Gamma$ on peut construire un isomorphisme entre les groupes (SO_3, \circ) et $(\mathbb{H}_1 / \mathcal{R}, *)$.

2.4.5 Angles d'Euler

On peut représenter une rotation par la composée de 3 rotations successives autour d'axes choisis arbitrairement : la rotation est alors représentée par les 3 angles de rotation.

Cette représentation a été utilisée par Euler pour poser les équations différentielles du mouvement d'un solide rigide.

Malheureusement elle ne possède pas de propriétés intéressantes pour ce qui est de la composition et de l'inversion des rotations. De plus rien qu'en se limitant aux 3 axes de coordonnées il y a déjà 12 façons de composer les rotations élémentaires (elles ne sont pas commutatives).

Nous n'en parlerons donc pas ici. On trouvera des algorithmes de conversion entre les angles d'Euler et les quaternions ou les matrices orthogonales dans l'article de Shoemake [100].

2.5 Passage d'une représentation à l'autre en dimension 3

On considère une rotation $r \in SO_3 \Gamma$ et ses différentes représentations vues dans la section précédente :

- sa matrice M dans la base canonique de $\mathbb{R}^3 \Gamma$

- son axe k et son angle θ tels que $r = r_1(k, \theta)$
- son vecteur rotation ω tel que $r = r_2(\omega)$
- son “logarithme” sous forme d’une matrice antisymétrique H vérifiant $M = \exp H$.
- un quaternion unitaire q tel que $r = r_3(q)$.

On veut construire des algorithmes permettant de passer d’une représentation à l’autre. La difficulté consiste à tenir compte des ambiguïtés et des cas particuliers inhérents à chaque représentation.

Ce sujet a été abordé par Shoemake dans [100] où il donne des algorithmes pour passer de M à q dans les deux sens. Dans le premier tome des “Graphics Gems” [55] Pique donne des algorithmes pour passer de (k, θ) à M dans les deux sens. Dornaïka [40] donne les formules de passage entre les différentes représentations (k, θ) , q et M . On trouve aussi des algorithmes de calcul de M et q en fonction de (k, θ) sur internet dans les *frequently asked questions (FAQ)* du groupe `comp.graphics.algorithms` (auteur non déterminé précisément).

Les trois représentations (k, θ) , ω et H sont trivialement reliées par les relations :

$$\begin{aligned}\omega &= k\theta, \\ H &= \tilde{\omega}.\end{aligned}\tag{2.31}$$

Le passage de ω à (k, θ) est ambigu : quand ω est nul on peut prendre k quelconque et θ nul et quand ω n’est pas nul on peut prendre indifféremment $(k, \theta) = (\frac{\omega}{|\omega|}, |\omega|)$ ou son opposé.

Ceci mis à part le problème se ramène finalement à trouver des algorithmes permettant de relier M et q et une de ces trois représentations “vectorielles”. Il n’y a donc que 6 transformations non triviales.

Dans les 6 paragraphes suivants nous résumons et comparons les différents algorithmes proposés. Au voisinage de l’identité et des retournements des instabilités apparaissent qui rendent nécessaire l’utilisation de développements limités. Certains de ceux présentés ici signalés par le symbole \blacktriangle constituent notre contribution.

2.5.1 $M \rightarrow$ vecteur

Les méthodes proposées utilisent la trace de M pour obtenir le cosinus de l’angle de la rotation et les éléments non diagonaux pour obtenir le sinus (comme il y a ambiguïté sur l’angle on le choisit entre 0 et π) :

$$\begin{aligned}\cos \theta &= \frac{\text{Tr } M - 1}{2}, \\ \sin \theta &= \frac{1}{2} \sqrt{(M_{21} - M_{12})^2 + (M_{13} - M_{31})^2 + (M_{32} - M_{23})^2}.\end{aligned}\tag{2.32}$$

Ceci définit dans tous les cas l’angle de la rotation.

Quand l'angle θ ainsi obtenu est assez éloigné de π on obtient les coordonnées du vecteur directeur unitaire k de l'axe par :

$$k_1 = \frac{M_{32} - M_{23}}{2 \sin \theta}, \quad k_2 = \frac{M_{13} - M_{31}}{2 \sin \theta}, \quad k_3 = \frac{M_{21} - M_{12}}{2 \sin \theta}. \quad (2.33)$$

Pour $\sin \theta$ proche de 0 Dornaïka [40] propose de choisir la plus grande valeur de la diagonale de M (supposons que ce soit M_{11}) et de calculer alors :

$$k_1 = \operatorname{sgn}(M_{32} - M_{23}) \sqrt{\frac{M_{11} - \cos \theta}{1 - \cos \theta}},$$

$$k_2 = \frac{M_{12} + M_{21}}{2k_1(1 - \cos \theta)}, \quad k_3 = \frac{M_{13} + M_{31}}{2k_1(1 - \cos \theta)}. \quad (2.34)$$

Les autres cas s'obtenant par permutation circulaire des indices. Cette méthode s'avère en fait très instable (une petite variation de M entraîne une grande variation de k) pour θ proche de 0. On ne l'utilisera donc qu'au voisinage de π ; les formules du paragraphe précédent restant assez stables en 0.

Au voisinage de $\theta = 0$ le vecteur rotation est préférable au couple (k, θ) et on prendra \blacktriangle :

$$\omega_1 = \frac{M_{32} - M_{23}}{2}, \quad \omega_2 = \frac{M_{13} - M_{31}}{2}, \quad \omega_3 = \frac{M_{21} - M_{12}}{2}. \quad (2.35)$$

2.5.2 $M \rightarrow q$

Les relations données par Shoemake dans [100] privilégient un ordre particulier dans l'évaluation des composantes de q . L'algorithme proposé par Dornaïka dans [40] est plus stable ; c'est celui que nous exposons ici.

On calcule tout d'abord les carrés des quatre composantes de $q = (s, u)$ en combinant les coefficients diagonaux de M :

$$\begin{aligned} 4s^2 &= 1 + M_{11} + M_{22} + M_{33}, \\ 4u_1^2 &= 1 + M_{11} - M_{22} - M_{33}, \\ 4u_2^2 &= 1 - M_{11} + M_{22} - M_{33}, \\ 4u_3^2 &= 1 - M_{11} - M_{22} + M_{33}. \end{aligned} \quad (2.36)$$

On peut ainsi en déduire la composante ayant la plus grande valeur absolue et calculer sa valeur en prenant la racine carrée du terme correspondant (ceci fixe le choix entre q et $-q$).

Les trois autres composantes de q se déduisent des six produits suivants obtenus à partir des coefficients non-diagonaux de M :

$$\begin{aligned} 4su_1 &= M_{32} - M_{23}, & 4su_2 &= M_{13} - M_{31}, & 4su_3 &= M_{21} - M_{12}, \\ 4u_2u_3 &= M_{32} + M_{23}, & 4u_3u_1 &= M_{13} + M_{31}, & 4u_1u_2 &= M_{21} + M_{12}. \end{aligned} \quad (2.37)$$

La composante connue intervient dans trois de ces produits qu'il suffit donc de diviser pour trouver les trois composantes manquantes.

2.5.3 $q \rightarrow$ vecteur rotation

Si $q = (s, u)$ calculer l'unique angle $\alpha \in [0, \pi]$ vérifiant $\cos \alpha = s$ et $\sin \alpha = |u|$ (par exemple en utilisant la fonction `atan2` de la librairie mathématique standard du langage \mathbb{C}).

Dans le cas où $|u|$ est petit (la rotation est proche de l'identité) prendre $\omega = 2u$
 ▲ sinon choisir $\omega = \frac{2\alpha}{|u|}u$.

2.5.4 $q \rightarrow M$

L'utilisation des quaternions unitaires permet d'obtenir ici une formule exempte de fonctions trigonométriques; en notant $q = (s, u)$ on a :

$$M = \begin{pmatrix} 1 - 2(u_2^2 + u_3^2) & 2(u_1u_2 - su_3) & 2(u_1u_3 + su_2) \\ 2(u_1u_2 + su_3) & 1 - 2(u_1^2 + u_3^2) & 2(u_2u_3 - su_1) \\ 2(u_1u_3 - su_2) & 2(u_2u_3 + su_1) & 1 - 2(u_1^2 + u_2^2) \end{pmatrix}. \quad (2.38)$$

2.5.5 Vecteur rotation $\rightarrow M$

Soit ω le vecteur de la rotation; en écrivant le développement de l'exponentielle et en identifiant les termes (cette technique est utilisée plus en détails dans la section suivante sur les rotations en dimension 4) on obtient :

$$M = \exp \tilde{\omega} = I_3 + \lambda(|\omega|)\tilde{\omega} + \mu(|\omega|)\tilde{\omega}^2, \quad (2.39)$$

en utilisant les fonctions (C^∞ en 0) :

$$\lambda(t) = \frac{\sin t}{t}, \quad \mu(t) = \frac{1 - \cos t}{t^2}. \quad (2.40)$$

Le cas $\omega = 0$ correspondant à $M = I_3$ mis à part en posant $(x, y, z) = \frac{\omega}{|\omega|}\theta = |\omega|\Gamma$
 $\lambda = \sin \theta$ et $\mu = 1 - \cos \theta$ on a donc :

$$M = \begin{pmatrix} 1 - \mu(1 - x^2) & \mu xy - \lambda z & \mu xz + \lambda y \\ \mu xy + \lambda z & 1 - \mu(1 - y^2) & \mu yz - \lambda x \\ \mu xz - \lambda y & \mu yz + \lambda x & 1 - \mu(1 - z^2) \end{pmatrix}. \quad (2.41)$$

Pour ω proche de 0 l'utilisation du développement limité de M en fonction de ω est préférable ▲ :

$$M = \frac{1}{2} \begin{pmatrix} 2 - \omega_2^2 - \omega_3^2 & \omega_1\omega_2 - 2\omega_3 & \omega_1\omega_3 + 2\omega_2 \\ \omega_1\omega_2 + 2\omega_3 & 2 - \omega_1^2 - \omega_3^2 & \omega_2\omega_3 - 2\omega_1 \\ \omega_1\omega_3 - 2\omega_2 & \omega_2\omega_3 + 2\omega_1 & 2 - \omega_1^2 - \omega_2^2 \end{pmatrix} + o(|\omega|^2) \quad (2.42)$$

2.5.6 Vecteur rotation $\rightarrow q$

On a la relation :

$$q = \left(\cos \frac{|\omega|}{2}, \frac{1}{2} \lambda \left(\frac{|\omega|}{2} \right) \omega \right), \quad (2.43)$$

dans laquelle λ est la fonction définie plus haut (sinus cardinal).

2.6 Les rotations en dimension 4

2.6.1 Introduction

Après avoir vu les rotations en dimension 2 et 3 on peut se demander à quoi ressemble le groupe SO_4 . Il se trouve que ce groupe occupe une place particulière parmi tous les groupes SO_n et présente quelques propriétés remarquables.

L'objectif de cette section est d'étudier les éléments de SO_4 en partant du fait que toute rotation est l'exponentielle d'un endomorphisme antisymétrique [85]. Nous verrons aussi le lien entre SO_4 et les quaternions unitaires.

Ce "logarithme" qui était l'angle en dimension 2 et le vecteur rotation (axe et angle) en dimension 3 est maintenant de dimension 6.

Les études de SO_4 dans les ouvrages de géométrie [17, 18, 38, 39, 85] portent d'une part sur les propriétés géométriques de ses éléments (sous-espaces invariants, angles, ...) et d'autre part sur sa structure (sous-groupes, décomposition en produits, isomorphismes, ...).

Les résultats présentés dans cette section correspondent à une étude fine des propriétés géométriques des rotations en dimension 4 qui n'est pas abordée dans les ouvrages précités.

Comme préliminaire nous pouvons reprendre le résultat de la section 2.2.1 qui nous permet d'affirmer que le triplet (p, q, r) associé à un élément de SO_4 ne peut prendre qu'une des valeurs suivantes : $(4, 0, 0)$ (identité), $(2, 2, 0)$, $(0, 4, 0)$ (symétrie centrale), $(2, 0, 1)$, $(0, 2, 1)$ et $(0, 0, 2)$. Pour toute rotation de SO_4 il existera une base orthonormée dans laquelle sa matrice sera :

$$R(\omega_1, \omega_2) = \begin{pmatrix} \cos \omega_1 & -\sin \omega_1 & 0 & 0 \\ \sin \omega_1 & \cos \omega_1 & 0 & 0 \\ 0 & 0 & \cos \omega_2 & -\sin \omega_2 \\ 0 & 0 & \sin \omega_2 & \cos \omega_2 \end{pmatrix}. \quad (2.44)$$

Si cette base n'est pas directe un échange des deux derniers vecteurs la rendra directe et changera la matrice de la rotation en $R(\omega_1, -\omega_2)$. Pour toute rotation de SO_4 il existe donc une **base orthonormée directe** de \mathbb{R}^4 dans laquelle sa matrice est $R(\omega_1, \omega_2)$.

Outre l'identité ($\omega_1 = \omega_2 = 0$) et la symétrie centrale ($\omega_1 = \omega_2 = \pi$) les rotations telles que $\omega_1 = \pm\omega_2$ forment deux ensembles remarquables que nous étudions en détail.

Notre point de départ sera le résultat du paragraphe 2.2.3. Nous allons donc prendre un endomorphisme antisymétrique et expliciter son exponentielle et en chercher les propriétés.

2.6.2 Polynôme caractéristique

Soit H la matrice antisymétrique de A_4 :

$$H = \begin{pmatrix} 0 & a & b & c \\ -a & 0 & d & e \\ -b & -d & 0 & f \\ -c & -e & -f & 0 \end{pmatrix}. \quad (2.45)$$

Nous allons voir comment on peut calculer explicitement l'exponentielle de $H\Gamma$ à partir de la connaissance de son polynôme minimal.

Ceci fait Γ nous verrons les liens entre les éléments géométriques de la rotation $\exp H$ et la matrice H .

Après quelques calculs Γ en posant :

$$\alpha = a^2 + b^2 + c^2 + d^2 + e^2 + f^2, \quad (2.46)$$

$$\beta = af - be + cd, \quad (2.47)$$

$$P(X) = X^2 + \alpha X + \beta^2, \quad (2.48)$$

on obtient la relation suivante :

$$P(H^2) = 0. \quad (2.49)$$

$P(X^2)$ est en fait le polynôme caractéristique de H .

Le cas $\alpha = 0$ correspond à $H = 0$ et donc $\exp H = I_4\Gamma$ qui ne nécessite pas d'étude particulière. Dans la suite Γ nous supposons donc que $\alpha > 0$.

Le discriminant de l'équation du second degré $P(x) = 0$ est

$$\Delta = \alpha^2 - 4\beta^2, \quad (2.50)$$

qui s'écrit encore

$$\Delta = \left[(a-f)^2 + (b+e)^2 + (c-d)^2 \right] \left[(a+f)^2 + (b-e)^2 + (c+d)^2 \right]. \quad (2.51)$$

Nous sommes ainsi naturellement amenés à considérer les deux cas $\Delta = 0$ et $\Delta > 0$. En effet Γ pour $\Delta = 0$ P a une racine double Γ et il s'avère que $P(X^2)$ n'est pas le polynôme minimal de H .

2.6.3 Cas $\Delta = 0$

D'après la relation 2.51 Γ Δ est nul quand H prend une des deux formes suivantes (la position un peu inattendue des signes trouvera une explication plus loin) :

$$H_g(u) = \begin{pmatrix} 0 & -u_1 & -u_2 & -u_3 \\ u_1 & 0 & -u_3 & u_2 \\ u_2 & u_3 & 0 & -u_1 \\ u_3 & -u_2 & u_1 & 0 \end{pmatrix}, \quad (2.52)$$

$$H_d(u) = \begin{pmatrix} 0 & u_1 & u_2 & u_3 \\ -u_1 & 0 & -u_3 & u_2 \\ -u_2 & u_3 & 0 & -u_1 \\ -u_3 & -u_2 & u_1 & 0 \end{pmatrix}, \quad (2.53)$$

avec $u \in \mathbb{R}^3$. Dans ce cas Γ H vérifie :

$$H^2 + \omega^2 I_4 = 0, \quad (2.54)$$

avec $\omega = |u|$.

Le polynôme $X^2 + \omega^2$ est dans ce cas le polynôme minimal de H .

Calcul de l'exponentielle

L'équation 2.54Γ analogue au cas général des rotations bidimensionnellesΓ permet d'en déduire facilement $\exp H\Gamma$ en remarquant qu'on peut exprimer toutes les puissances de H comme combinaisons linéaires de I_4 et H :

$$H^{2p} = (-1)^p \omega^{2p} I_4, \quad (2.55)$$

$$H^{2p+1} = (-1)^p \omega^{2p} H. \quad (2.56)$$

En identifiant les développements en série entièreΓ on obtient :

$$\exp H = (\cos \omega) I_4 + \frac{\sin \omega}{\omega} H. \quad (2.57)$$

Lien avec les quaternions unitaires

Etant donné un quaternion $q = (s, u)$ Γ l'application qui à un quaternion r associe son produit à gauche par q (soit $r \mapsto q * r$) est linéaire :

$$q * (\lambda_1 r_1 + \lambda_2 r_2) = \lambda_1 (q * r_1) + \lambda_2 (q * r_2). \quad (2.58)$$

La matrice $Q_g(q)$ dans la base $(1, i, j, k)$ de cette application est :

$$Q_g(q) = \begin{pmatrix} s & -u_1 & -u_2 & -u_3 \\ u_1 & s & -u_3 & u_2 \\ u_2 & u_3 & s & -u_1 \\ u_3 & -u_2 & u_1 & s \end{pmatrix}. \quad (2.59)$$

On a ainsi par définition :

$$\forall q, r \in \mathbb{H}, q * r = Q_g(q)r. \quad (2.60)$$

En outreΓ $Q_g\Gamma$ définie de \mathbb{H} dans $M_4\Gamma$ possède les propriétés suivantes vis-à-vis des opérations définies dans \mathbb{H} (leur démonstration n'est qu'un "jeu d'écriture") :

$$\begin{aligned} Q_g(q + q') &= Q_g(q) + Q_g(q'), \\ Q_g(q * q') &= Q_g(q)Q_g(q'), \\ Q_g(\bar{q}) &= Q_g(q)^T, \\ Q_g(q)Q_g(q)^T &= |q|^2 I_4, \end{aligned} \quad (2.61)$$

ceci pour tout $q, q' \in \mathbb{H}$.

On peut de la même façon définir l'application de multiplication à droite par le conjugué d'un quaternion $r = (t, v)$:

$$\begin{aligned} \mathbb{H} &\rightarrow \mathbb{H} \\ q &\mapsto q * \bar{r}. \end{aligned} \quad (2.62)$$

L'utilisation du conjugué de r au lieu de r lui-même permet de conserver les mêmes propriétés que Q_g pour la multiplication. Cette application est linéaireΓ sa matrice

dans la base $(1, i, j, k)$ étant :

$$Q_d(r) = \begin{pmatrix} t & v_1 & v_2 & v_3 \\ -v_1 & t & -v_3 & v_2 \\ -v_2 & v_3 & t & -v_1 \\ -v_3 & -v_2 & v_1 & t \end{pmatrix}. \quad (2.63)$$

On a des propriétés identiques à celles vues plus haut pour $Q_g \Gamma$ valables pour tout $r, r' \in \mathbb{H}$:

$$\begin{aligned} r' * \bar{r} &= Q_d(r)r', \\ Q_d(r + r') &= Q_d(r) + Q_d(r'), \\ Q_d(r * r') &= Q_d(r)Q_d(r'), \\ Q_d(\bar{r}) &= Q_d(r)^T, \\ Q_d(r)Q_d(r)^T &= |r|^2 I_4. \end{aligned} \quad (2.64)$$

On a alors les propriétés remarquables suivantes :

$$\begin{aligned} \forall u \in \mathbb{R}^3, \exp H_g(u) &= Q_g(q), \\ \forall u \in \mathbb{R}^3, \exp H_d(u) &= Q_d(q), \end{aligned} \quad (2.65)$$

avec $q = (\cos|u|, \frac{\sin|u|}{|u|}u) \in \mathbb{H}_1$.

L'ensemble des rotations formé par les exponentielles des matrices de type H_g (resp. H_d) est exactement l'ensemble des matrices $Q_g(q)$ (resp. $Q_d(q)$) avec $q \in \mathbb{H}_1$.

Il n'y a pas unicité dans l'autre sens : pour un quaternion unitaire q il existe une infinité de vecteurs $u \in \mathbb{R}^3$ vérifiant les relations 2.65. Par contre pour les rotations différentes de $-Id \Gamma$ on peut imposer la condition supplémentaire $u \in B_3(O, \pi)$ qui garantit alors l'unicité de u .

Définissons les deux parties de SO_4 :

$$R_g = \left\{ \exp H_g(u), u \in \mathbb{R}^3 \right\}, \quad (2.66)$$

$$R_d = \left\{ \exp H_d(u), u \in \mathbb{R}^3 \right\}. \quad (2.67)$$

$Q_g : \mathbb{H}_1 \rightarrow SO_4$ est un morphisme de groupes d'après 2.61 ; son image R_g est donc un sous-groupe de SO_4 . Q_g étant injectif il établit un isomorphisme entre les groupes $(\mathbb{H}_1, *)$ et (R_g, \circ) . On a de même un isomorphisme entre $(\mathbb{H}_1, *)$ et (R_d, \circ) .

Eléments géométriques

L'expression 2.57 permet de calculer l'image d'un vecteur $x \in \mathbb{R}^4$ par la rotation $\exp H$:

$$(\exp H)x = \cos \omega x + \sin \omega \frac{Hx}{\omega}. \quad (2.68)$$

On peut montrer en tirant parti des relations $H^T = -H$ et $H^2 = -\omega^2 I_4$ que le vecteur $\frac{Hx}{\omega}$ est orthogonal à x et de même norme. Ainsi chaque vecteur non nul x

effectue une rotation d'angle ω dans le plan dont une base orthogonale directe est $(x, \frac{Hx}{\omega})$. Ce plan est donc globalement invariant par $\exp H$.

Si on choisit un vecteur unitaire x puis un vecteur unitaire y orthogonal à x et $\frac{Hx}{\omega}$ le vecteur $\frac{Hy}{\omega}$ est alors non seulement orthogonal à y comme on l'a vu plus haut mais aussi à x et à $\frac{Hx}{\omega}$. Il existe donc une infinité de décompositions de \mathbb{R}^4 en somme directe orthogonale de 2 plans vectoriels globalement invariants par $\exp H$ et sur lesquels la restriction de $\exp H$ est une rotation d'angle ω .

2.6.4 Cas $\Delta > 0$

Dans ce cas P a deux racines réelles négatives distinctes (leur produit étant $\beta^2 \geq 0$ et leur somme $-\alpha < 0$) que nous noterons $-\omega_1^2$ et $-\omega_2^2$:

$$\omega_1 = \sqrt{\frac{\alpha + \sqrt{\Delta}}{2}}, \omega_2 = \sqrt{\frac{\alpha - \sqrt{\Delta}}{2}}. \quad (2.69)$$

Calcul de l'exponentielle

La relation 2.48 qui s'écrit aussi $H^4 = -\alpha H^2 - \beta^2 I_4$ permet d'exprimer $\exp H$ comme combinaison linéaire dans la base (I_4, H, H^2, H^3) . Après écriture de la série et identification des différents termes on obtient :

$$\begin{aligned} \exp H &= \left(1 - \frac{\beta^2}{\sqrt{\Delta}} (\lambda_0(\omega_2) - \lambda_0(\omega_1))\right) I_4 \\ &+ \left(1 - \frac{\beta^2}{\sqrt{\Delta}} (\lambda_1(\omega_2) - \lambda_1(\omega_1))\right) H \\ &+ \frac{1}{\sqrt{\Delta}} (\lambda_2(\omega_2) - \lambda_2(\omega_1)) H^2 \\ &+ \frac{1}{\sqrt{\Delta}} (\lambda_3(\omega_2) - \lambda_3(\omega_1)) H^3, \end{aligned} \quad (2.70)$$

En posant :

$$\lambda_0(t) = \frac{1 - \cos t}{t^2}, \quad \lambda_1(t) = \frac{t - \sin t}{t^3}, \quad \lambda_2(t) = \cos t, \quad \lambda_3(t) = \frac{\sin t}{t}. \quad (2.71)$$

Eléments géométriques

On a la factorisation suivante :

$$(H^2 + \omega_1^2 I_4)(H^2 + \omega_2^2 I_4) = 0. \quad (2.72)$$

Soit $x \in \mathbb{R}^4$ un élément du noyau de $H^2 + \omega_1^2 I_4$ vérifiant donc $H^2 x = -\omega_1^2 x$. Ce noyau est alors de dimension 2 et engendré par les deux vecteurs orthogonaux x et Hx . De plus pour tout entier p $H^p x$ est aussi dans ce noyau donc $(\exp H)x$ aussi et on a :

$$(\exp H)x = \cos \omega_1 x + \sin \omega_1 Hx. \quad (2.73)$$

$\ker(H^2 + \omega_1^2 I_4)$ est donc un plan globalement invariant par $\exp H$ sur lequel sa restriction est une rotation d'angle ω_1 dans n'importe quelle base (x, Hx) . De même le plan $\ker(H^2 + \omega_2^2 I_4)$ est globalement invariant par $\exp H$ et sa restriction est une rotation d'angle ω_2 dans toute base du type (x, Hx) .

2.7 Conclusion

Dans ce chapitre nous avons rappelé les définitions relatives aux rotations en dimension quelconque leurs propriétés (exponentielle sous-espaces invariants décomposition en produit de réflexions) ainsi que les concepts liés aux rotations en dimensions 2 et 3 (nombres complexes quaternions).

Nous avons ensuite étudié les différentes représentations des rotations en dimensions 2 et 3 ainsi que les relations entre ces représentations. En particulier dans le cas du passage d'une représentation à l'autre en dimension 3 nous avons donné quelques nouvelles expressions rendant les calculs plus robustes au voisinage des singularités des représentations.

Enfin nous avons étudié les propriétés géométriques des rotations en dimension 4. Ceci nous a permis de voir le lien entre le groupe des quaternions unitaires et les deux sous-groupes distingués non triviaux de SO_4 . Les matrices Q_d et Q_g introduites à cette occasion serviront dans le chapitre 4 pour résoudre le problème de la mise en correspondance de deux nuages de points appariés dans laquelle la partie linéaire (une rotation de SO_3) du déplacement inconnu est représentée par un quaternion unitaire.

Nous pensons d'autre part que l'étude de SO_4 peut avoir des applications en vision les matrices des transformations projectives étant dans M_4 .

Chapitre 3

Modélisation du comportement des solides déformables

Dans ce chapitre essentiellement bibliographique nous présentons la mécanique des milieux continus, la modélisation du comportement élastique, les déformations non réversibles (plasticité, fractures). Nous voyons ensuite comment ces équations sont discrétisées puis résolues sur ordinateur.

Nous adoptons l'optique des ingénieurs prédisant le comportement des structures. Elle est orientée vers le réalisme physique et la précision des résultats, plutôt que le réalisme visuel et la rapidité du calcul, comme l'est la synthèse d'animations par ordinateur.

3.1 Mécanique des milieux continus

Cette section est basée sur différents cours de physique [81] [37], livres de mécanique générale [53] [54] et ouvrages portant spécifiquement sur la mécanique des milieux continus et l'élasticité [30] [16] [43]. La démarche et les notations adoptées sont celles de Ciarlet dans [30].

Le terme **milieu continu** désigne un système de particules suffisamment dense pour pouvoir être décrit par des grandeurs continues et dont les particules conservent en moyenne les mêmes positions relatives au cours du temps. Les liquides et les gaz dans les conditions usuelles, ainsi que les solides rigides ou déformables, sont des milieux continus.

Notre intérêt porte essentiellement sur les solides de dimension 3, mais la dimension 2 permet d'illustrer plus facilement les différentes notions. La dimension de l'espace sera donc désignée par d , valant 2 ou 3. Les figures illustreront, sauf exception, le cas $d = 2$.

Dans ce qui suit, l'évolution du système au cours du temps se fait de façon **quasi-statique** : à tout moment, le système est dans un état d'équilibre statique.

Nous définissons d'abord une représentation continue d'un solide et de sa déformation. Après avoir étudié localement les propriétés de la déformation, nous verrons comment s'écrit l'équilibre du solide. Ensuite, nous présenterons différents modèles

de comportement (quelles sont les contraintes internes provoquées par une déformation du solide).

3.1.1 Déformation

Un solide déformé sera représenté par un domaine $\Omega \subset \mathbb{R}^d$ et une fonction $\varphi : \bar{\Omega} \rightarrow \mathbb{R}^d$. $\bar{\Omega}$ est l'espace occupé par l'ensemble des points du solide dans une configuration arbitraire dite **configuration de référence**. Usuellement on choisit une configuration dans laquelle le solide est à l'équilibre et n'est soumis à aucune contrainte extérieure mais ce n'est pas indispensable. φ est appelée la **déformation** ; elle associe à un point $x \in \bar{\Omega}$ de la configuration de référence sa position dans la configuration déformée $x^\varphi = \varphi(x)$. La notation G^φ désigne une grandeur G mesurée dans la configuration déformée. On notera ainsi Ω^φ l'image de Ω par φ et Γ^φ la frontière de Ω^φ .

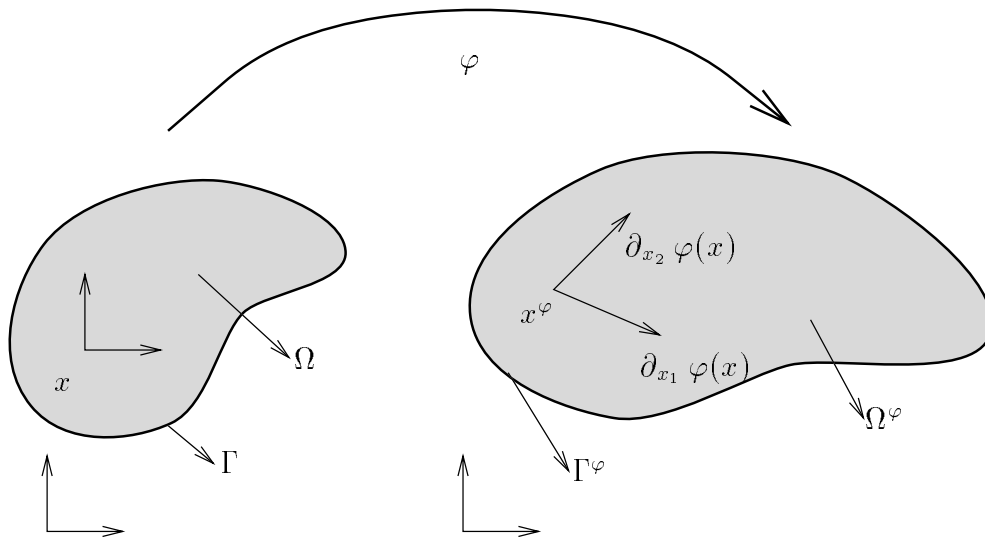


Fig. 41 - Déformation d'un objet.

On supposera que φ est suffisamment régulière pour que les dérivées utilisées dans la suite soient définies.

3.1.2 Étude locale des déformations

Condition de préservation de l'orientation

On impose à la déformation de préserver l'orientation du solide :

$$\forall x \in \Omega, \det \nabla \varphi(x) > 0. \quad (3.1)$$

$\nabla \varphi(x)$ désigne le gradient de φ en x ; c'est une matrice $d \times d$:

$$\nabla \varphi(x) = \begin{pmatrix} \partial_{x_1} \varphi_1(x) & \dots & \partial_{x_d} \varphi_1(x) \\ \vdots & \ddots & \vdots \\ \partial_{x_1} \varphi_d(x) & \dots & \partial_{x_d} \varphi_d(x) \end{pmatrix}. \quad (3.2)$$

La condition 3.1 impose de plus à la déformation de ne pas avoir de points singuliers (vérifiant $\det \nabla \varphi(x) = 0$) pour lesquels la matière serait infiniment comprimée ce qui n'est pas physiquement réaliste.

Approximation locale

On a une approximation de φ au premier ordre :

$$\varphi(x + \delta x) = \varphi(x) + \nabla \varphi(x) \delta x + o(\|\delta x\|). \quad (3.3)$$

Parties rigide et non rigide de la déformation

On sait (**factorisation polaire**) que toute matrice inversible M s'écrit de manière unique $M = QS$ où Q est une matrice orthogonale et S une matrice symétrique définie positive.

Dans le cas de $\nabla \varphi$ on a

$$\nabla \varphi(x) = Q_\varphi(x) S_\varphi(x), \quad (3.4)$$

$Q_\varphi(x)$ étant de plus une rotation (une matrice orthogonale de déterminant $+1$) puisque nous avons imposé la condition $\det \nabla \varphi(x) > 0$.

La dépendance de S_φ en fonction de $\nabla \varphi$ n'étant pas "simple" on écrira souvent les équations en fonction de

$$C(x) = S_\varphi(x)^2 = \nabla \varphi(x)^T \nabla \varphi(x). \quad (3.5)$$

La matrice symétrique $C(x)$ est appelée **tenseur des déformations de Cauchy–Green**.

$Q_\varphi(x)$ correspond à la partie rigide de la déformation en x . La matrice $S_\varphi(x)$ décrit la partie non rigide de la déformation.

Dans le cas d'une déformation rigide (la composée d'une translation et d'une rotation) $S_\varphi(x)$ est égale à l'identité en tout point $x \in \Omega$. Réciproquement si $S_\varphi(x)$ est l'identité en tout point $x \in \Omega$ alors la déformation est rigide.

Cette propriété conduit à la définition de la matrice symétrique

$$E(x) = \frac{1}{2} (C(x) - I), \quad (3.6)$$

appelée **tenseur des déformations de Green–Lagrange**. $E(x)$ est nul quand la déformation est "localement rigide" en x . On peut montrer que

$$S_\varphi = I + E + o(\|E\|). \quad (3.7)$$

Variation de longueur

On considère un "petit" vecteur δx en un point x . On se demande quelle est la variation de longueur de son image $\delta x^\varphi = \varphi(x + \delta x) - \varphi(x)$. On a :

$$\|\delta x^\varphi\|^2 = \delta x^T S_\varphi^2(x) \delta x + o(\|\delta x\|^2), \quad (3.8)$$

$$= \delta x^T C(x) \delta x + o(\|\delta x\|^2). \quad (3.9)$$

Variation de volume

On considère un élément de volume (élément de surface pour $d = 2$) dv en un point x . Le volume de son image dv^φ est

$$dv^\varphi = \det S_\varphi(x) dv \quad (3.10)$$

$$= \det \nabla \varphi(x) dv. \quad (3.11)$$

Si on note $\rho(x)$ la masse volumique du solide en un point x de sa configuration de référence Γ alors la masse volumique $\rho^\varphi(x^\varphi)$ au point x^φ de la configuration déformée vérifie

$$\rho^\varphi(x^\varphi) = \frac{1}{\det \nabla \varphi(x)} \rho(x). \quad (3.12)$$

3.1.3 Lois mécaniques d'équilibre

On suppose le solide à l'équilibre dans une configuration déformée. Il est soumis à un champ de forces volumiques f^φ défini sur Ω^φ et sur une partie $\Gamma_1^\varphi \subset \Gamma^\varphi$ de sa frontière Γ à un champ de forces de surface g^φ .

On cherche quelles sont les contraintes internes au solide qui s'opposent à ces contraintes extérieures pour maintenir l'équilibre.

Plaçons-nous en un point x^φ du solide déformé. Choisissons un plan arbitraire passant par ce point Γ défini par un vecteur normal unitaire n^φ et imaginons que l'on retire une portion de matière au dessus de ce plan au voisinage du point x^φ . Il faut alors appliquer une certaine force par unité de surface en x^φ pour remplacer celle qu'exerçait la portion de solide retirée. C'est cette force qui représente les contraintes internes au solide. Elle est définie de manière analogue à la tension d'une corde. Notons qu'elle n'est en général pas colinéaire à n^φ sauf pour un nombre fini de directions.

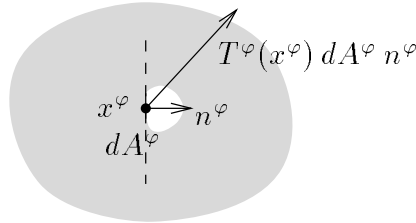


Fig. 42 - Contraintes internes en un point x^φ .

On peut montrer (**théorème de Cauchy**) que cette force dépend linéairement de n^φ . Cette application linéaire est représentée par une matrice $T^\varphi(x^\varphi)$ de dimensions $d \times d$ appelée le **tenseur des contraintes de Cauchy**.

Le fait que le solide soit en équilibre permet d'établir les relations

$$\begin{cases} -\operatorname{div}^\varphi T^\varphi = f^\varphi & \text{dans } \Omega^\varphi \\ T^\varphi = T^{\varphi T} & \text{dans } \Omega^\varphi \\ T^\varphi n^\varphi = g^\varphi & \text{sur } \Gamma_1^\varphi \end{cases} \quad (3.13)$$

Dans la plupart des cas $\Gamma\varphi$ est l'inconnue. Il est alors intéressant de "déplacer" ces équations dans la configuration de référence Γ en définissant les grandeurs suivantes :

$$\Sigma(x) = \det \nabla \varphi(x) \nabla \varphi(x)^{-1} T^\varphi(x^\varphi) \nabla \varphi(x)^{-T}, \quad (3.14)$$

$$f(x) = \det \nabla \varphi(x) f^\varphi(x^\varphi), \quad (3.15)$$

$$g(x) = \det \nabla \varphi(x) |\nabla \varphi(x)^{-T} n| g^\varphi(x^\varphi). \quad (3.16)$$

$\Sigma(x)$ et $f(x)$ sont définies pour $x \in \Omega$ et $g(x)$ pour $x \in \Gamma_1$.

Σ est appelé le **second tenseur des contraintes de Piola–Kirchhoff** ; il est en fait défini à partir du premier tenseur des contraintes de Piola–Kirchhoff T qui vaut $T(x) = \nabla \varphi \Sigma(x)$. $T(x)$ est défini de telle sorte que sa divergence soit colinéaire à celle de T^φ et que pour un élément d'aire quelconque dS on ait $T dS = T^\varphi dS^\varphi$.

En utilisant ces nouvelles grandeurs Γ le système 3.13 s'écrit alors

$$\begin{cases} -\operatorname{div}(\nabla \varphi \Sigma) = f & \text{dans } \Omega \\ \Sigma = \Sigma^T & \text{dans } \Omega \\ (\nabla \varphi \Sigma) n = g & \text{sur } \Gamma_1 \end{cases} \quad (3.17)$$

On écrit cette équation en fonction de Σ plutôt que de T pour conserver la condition de symétrie (que T ne vérifie pas).

On peut écrire chacun de ces deux systèmes sous une forme variationnelle formellement équivalente. Dans ce qui suit $\Gamma * \theta$ désigne le produit scalaire "terme à terme" de deux matrices $d \times d$: $A * B = \sum_{i,j} A_{ij} B_{ij}$ et $\Gamma \cdot \theta$ désigne le produit scalaire de deux vecteurs de \mathbb{R}^d : $u \cdot v = \sum_i u_i v_i$.

Dans la configuration déformée Γ^φ le système 3.13 est équivalent à :

$$\int_{\Omega^\varphi} T^\varphi * \nabla \theta^\varphi dx^\varphi = \int_{\Omega^\varphi} f^\varphi \cdot \theta^\varphi dx^\varphi + \int_{\Gamma_1^\varphi} g^\varphi \cdot \theta^\varphi dS^\varphi \quad (3.18)$$

Pour tout champ de vecteurs $\theta^\varphi : \Omega^\varphi \rightarrow \mathbb{R}^d$
suffisamment régulier et nul sur $\Gamma^\varphi - \Gamma_1^\varphi$.

Dans la configuration de référence Γ le système 3.17 est équivalent à :

$$\int_{\Omega} (\nabla \varphi \Sigma) * \nabla \theta dx = \int_{\Omega} f \cdot \theta dx + \int_{\Gamma_1} g \cdot \theta dS \quad (3.19)$$

Pour tout champ de vecteurs $\theta : \Omega \rightarrow \mathbb{R}^d$
suffisamment régulier et nul sur $\Gamma - \Gamma_1$.

Les termes correspondant aux travaux des forces internes ont même valeur :

$$\int_{\Omega^\varphi} T^\varphi * \nabla \theta^\varphi dx^\varphi = \int_{\Omega} (\nabla \varphi \Sigma) * \nabla \theta dx. \quad (3.20)$$

Le système 3.18 (resp. 3.19) est appelé **principe des travaux virtuels** dans la configuration déformée (resp. dans la configuration de référence).

Ce principe est déduit ici du système d'équations aux dérivées partielles (e.d.p.) à partir de manipulations algébriques. Ce système d'e.d.p. découle lui-même de l'écriture de la condition d'équilibre d'un petit volume de solide autour d'un point en utilisant l'axiome "relation fondamentale de la dynamique".

On peut aussi se baser sur l'axiome "principe des travaux virtuels" : dans une position d'équilibre Γ pour une déformation virtuelle autour de cette position Γ la somme des travaux des forces intérieures égale la somme des travaux des forces extérieures. Dans ce cas Γ on aurait d'abord obtenu les conditions 3.18 et 3.19 puis on en aurait déduit 3.13 et 3.17.

3.2 Comportement des solides

3.2.1 Nécessité d'une loi de comportement

Qu'il soit formulé dans la configuration déformée ou dans la configuration de référence Γ sous forme différentielle ou sous forme variationnelle Γ ce système ne fournit pas assez d'équations pour permettre de trouver les positions d'équilibre.

Il faut lui adjoindre une loi associant les contraintes aux déformations. Cette loi modélise la réponse du matériau Γ tout comme la relation

$$Tension = k \text{ Allongement}$$

modélise la réponse élastique d'un ressort et doit être ajoutée à la relation fondamentale de la dynamique pour permettre le calcul de la position d'équilibre d'une masse ponctuelle suspendue à un ressort.

Cette loi doit donner une valeur aux contraintes en fonction de la déformation. Les contraintes ne doivent pas dépendre de la partie rigide de la déformation (le comportement du solide ne dépend pas du repère choisi pour le décrire). La loi de comportement peut tenir compte de l'homogénéité du matériau et être alors la même en tout point du solide. Elle peut aussi tenir compte de son isotropie (réponse identique dans toutes les directions).

3.2.2 Lois usuelles

Loi de Hooke

La loi de comportement la plus simple est obtenue en faisant les hypothèses d'homogénéité du matériau Γ d'isotropie du comportement et de la dépendance linéaire entre Σ (représentant les contraintes dans le matériau) et E (représentant la partie non-rigide de la déformation).

Ces hypothèses s'appliquent bien à tous les matériaux "durs" (métaux Γ rocs Γ verre Γ bois Γ ...) pour des déformations suffisamment petites.

Sous ces hypothèses Γ la loi de comportement s'écrit :

$$\Sigma = \Sigma_0 + \lambda \text{ Tr } E I + 2\mu E, \quad (3.21)$$

et ne dépend que de deux paramètres λ et μ appelés **coefficients de Lamé** du matériau. Cette relation porte le nom de **loi de Hooke**. Σ_0 représente les contraintes présentes dans le matériau dans la configuration de référence.

Notons que si Σ dépend linéairement de E Γ il ne dépend pas linéairement des coefficients du gradient de la déformation.

Les coefficients de Lamé ne sont pas mesurables directement. On leur préfère le couple équivalent de coefficients \mathcal{E} et ν appelés respectivement **module d'Young** et **coefficient de Poisson** du matériau. \mathcal{E} est l'équivalent de la raideur d'un ressort : on peut le mesurer en étirant un petit cylindre allongé et homogène du matériau. Quand on exerce une traction axiale sur les extrémités de ce cylindre on observe (quel que soit le matériau) que son diamètre diminue au fur et à mesure qu'il s'allonge ; ν représente le rapport de la diminution relative de son diamètre et de l'augmentation relative de sa longueur.

On trouvera des tableaux donnant les valeurs de \mathcal{E} et ν pour quelques matériaux usuels dans [16] (section 4.4) [43] (section 6.3) et dans [30] (section 1.3).

λ et μ se déduisent de \mathcal{E} et ν par les relations suivantes :

$$\lambda = \frac{\mathcal{E}\nu}{(1-2\nu)(1+\nu)}, \quad (3.22)$$

$$\mu = \frac{\mathcal{E}}{1+\nu}. \quad (3.23)$$

Hyperélasticité

On peut aussi définir en chaque point du solide une **densité volumique d'énergie potentielle élastique** $W \in \mathbb{R}$ dépendant de x et $E(x)$ on définit alors la loi de comportement

$$\Sigma = \frac{\partial W(x, E(x))}{\partial E}. \quad (3.24)$$

W est bien une densité d'énergie : sa variation lors d'une petite déformation correspond au travail des forces élastiques.

Un tel comportement s'appelle **hyperélasticité** ; le matériau est dit hyperélastique.

La loi de Hooke 3.21 dérive de la densité

$$W = \frac{\lambda}{2}(\text{Tr } E)^2 + \mu \text{Tr}(E^2). \quad (3.25)$$

Dans le cas d'un matériau hyperélastique on peut définir l'énergie potentielle élastique du solide :

$$I = \sum_{\Omega} W(x, E(x)) dx. \quad (3.26)$$

Les termes correspondant aux travaux des forces internes dans les équations 3.18 et 3.19 peuvent être remplacés par les variations de cette énergie potentielle en fonction du déplacement virtuel θ .

3.2.3 Comportements limites

Pour de petites déformations le solide retourne toujours dans la même configuration après la suppression des contraintes qui les entraînent.

Des déformations plus importantes entraînent une modification instantanée de cette configuration au repos : on parle de **plasticité**.

Des déformations encore plus importantes provoquent l'apparition de failles et puis la séparation du solide en plusieurs parties : on parle de **fracture**.

Dans certains cas et comme par exemple les métaux en fusion la configuration au repos change au cours du temps et évolue vers la configuration courante.

Finalement il peut arriver qu'une position d'équilibre soit instable et il y a alors possibilité de "saut" instantané vers une autre position d'équilibre parfois très éloignée de la précédente. On parle de "flambage". Ce phénomène se produit en particulier pour des objets minces (plaques, tiges, ...).

Il existe un grand nombre de conditions prédisant l'apparition de ces phénomènes et de méthodes pour les modéliser.

3.3 Discrétisation et résolution

Une recherche informatique des positions d'équilibre d'un solide ou d'un système de solides passe par une représentation discrète des inconnues et en l'occurrence de la déformation φ .

La méthode utilisée dans la grande majorité des cas est la **méthode des éléments finis**. Elle consiste à partitionner le solide en briques de base (cubes ou tétraèdres déformés) jointives. La déformation est définie par morceaux sur chacune de ces briques et en s'arrangeant pour que le recollement s'effectue avec la continuité recherchée. Les inconnues du problème discrétisé sont alors les paramètres de la déformation.

On introduit cette représentation dans une des deux formulations variationnelles 3.18 ou 3.19. Ceci fournit pour chaque déplacement virtuel θ une équation portant sur les paramètres définissant la déformation.

Un choix judicieux de n (le nombre d'inconnues) fonctions θ fournit alors un système $n \times n$ que l'on résout soit par des méthodes directes comme le pivot de Gauss soit par des méthodes itératives.

Nous n'irons pas plus loin ici dans la description de la méthode des éléments finis : elle est très clairement exposée dans [118, 116, 117] et ainsi que dans [15]. L'ensemble de ces 3 livres abondamment illustrés d'exemples de différents problèmes physiques permet d'aborder facilement les différents aspects de cette méthode.

Dans le chapitre 3 de la partie *Animation de systèmes de solides* nous voyons en détail quelques aspects de la méthode des éléments finis.

Chapitre 4

Mise en correspondance

La mise en correspondance est un problème d'optimisation dans lequel on cherche à superposer deux structures ou une structure avec plusieurs autres. Ces structures sont en général issues du même objet et acquises à travers des modalités différentes.

Le terme structure désigne par exemple :

- un nuage de points correspondant aux positions de points caractéristiques d'un objet mesurées grâce à un localisateur tridimensionnel (optique ou électromagnétique ou échographie 1D)
- un échantillon de la surface d'un objet obtenu par des appareils de balayage tridimensionnel (laser ou optique)
- un ensemble de points résultant d'une procédure d'extraction de contours ou de points caractéristiques (voir le chapitre 5)
- une image tridimensionnelle (scanner IRM...) ou bidimensionnelle (radio ou image vidéo ou échographie...) brute.

L'inconnue de la mise en correspondance est une déformation au sens défini dans le chapitre 3 de cette partie.

Après avoir vu dans la section 4.1 comment on peut formaliser ce problème et donné quelques références sur les travaux dans ce domaine nous consacrons la section 4.2 à l'étude d'un des problèmes les plus simples de mise en correspondance : la mise en correspondance de nuages de points appariés.

La résolution efficace et robuste de ce problème est une étape cruciale dans de nombreuses applications en particulier dans les capteurs de position optiques comme le système Optotrak(R). Ce problème sert en outre de composant de base dans la plupart des méthodes de mise en correspondance de structures plus complexes.

Pour ce problème nous verrons les différentes solutions proposées dans la littérature. Nous mettrons en lumière les liens entre ces solutions. Nous proposons aussi une solution nouvelle utilisant la représentation d'une rotation par son vecteur rotation.

4.1 Position du problème

Soient deux structures $S_{\mathcal{A}}$ et $S_{\mathcal{B}}$ appartenant au même objet Γ et connues dans deux repères \mathcal{A} et \mathcal{B} . On souhaite déformer le repère \mathcal{B} pour superposer au mieux les deux structures Γ ou plus précisément $\Gamma S_{\mathcal{A}}$ et l'image de $S_{\mathcal{B}}$ par la déformation.

Afin de mettre ce problème en équations Γ on définit une distance entre deux structures données dans \mathcal{A} Γ notée $D(S, S')$. On cherche alors une déformation φ dans un ensemble donné Φ . Φ est par exemple l'ensemble des déplacements rigides en dimension 3 Γ l'ensemble des similitudes directes en dimension 3 Γ une certaine classe de déformations non rigides.

L'imposition de contraintes sur φ se fait en introduisant une fonctionnelle $E(\varphi)$ minimale quand φ vérifie les contraintes. E peut par exemple être une énergie potentielle élastique (voir le chapitre 3 de cette partie) et contraindre ainsi φ à être la "plus rigide possible".

Le problème de la mise en correspondance est alors le problème d'optimisation suivant :

$$\begin{cases} \varphi \in \Phi, \\ D(S_{\mathcal{A}}, \varphi(S_{\mathcal{B}})) + E(\varphi) \quad \text{minimal} \end{cases}$$

4.2 Mise en correspondance de nuages de points appariés

Dans le problème de la mise en correspondance de nuages de points Γ les deux structures sont un même ensemble de n points Γ aux coordonnées mesurées dans deux repères différents. On se place dans le cas où les nuages de points sont appariés : on sait quel point de la première série de mesures correspond à tel autre point de la seconde.

4.2.1 Mise en équations

Comme nous l'avons dit dans l'introduction de ce chapitre Γ le problème de la mise en correspondance rigide de nuages de points appariés est important et a déjà été largement étudié. On le formalise le plus souvent ainsi :

Problème 1 (Mise en correspondance rigide de nuages de points appariés)

Etant données deux familles de vecteurs a_i et b_i ($i = 1 \dots n$) représentant les coordonnées du même point M_i d'un objet rigide dans deux repères orthonormés directs \mathcal{A} et \mathcal{B} Γ trouver la transformation rigide φ minimisant l'erreur aux moindres carrés :

$$\varepsilon^2(\varphi) = \mathcal{S} \left[\|a_i - \varphi(b_i)\|^2 \right]. \quad (4.1)$$

L'opérateur $\mathcal{S}[E_i]$ désigne la moyenne des grandeurs E_i pour $i = 1 \dots n$ Γ soit : $\mathcal{S}[E_i] = \frac{1}{n} \sum_{i=1}^n E_i$. On cherche ainsi à superposer l'image du nuage \mathcal{B} au nuage \mathcal{A} .

On peut introduire des poids dans la moyenne : chaque paire de points a une contribution différente dans ε . On peut aussi introduire un facteur d'échelle inconnu :

on recherche alors la meilleure similitude directe. Ces modifications du problème conduisent aux mêmes équations [64Γ65Γ70].

On notera dans la suite d la dimension de l'espace ; d valant usuellement 2 ou 3.

4.2.2 Problème linéaire associé

La transformation rigide φ est composée d'une translation $t \in \mathbb{R}^d$ et d'une rotation $R \in SO_d$:

$$\forall m \in \mathbb{R}^d, \varphi(m) = t + R(m). \quad (4.2)$$

Nous allons chercher la meilleure translation possible pour une rotation donnée. En différenciant la relation 4.1 par rapport à t pour R constant on obtient :

$$d(\varepsilon^2) = 2 \langle dt, t + R(\mathcal{S}[b_i]) - \mathcal{S}[a_i] \rangle. \quad (4.3)$$

A R fixé ε est donc minimal quand

$$t = \mathcal{S}[a_i] - R(\mathcal{S}[b_i]). \quad (4.4)$$

Cette translation est celle qui fait coïncider le barycentre de l'image du nuage de \mathcal{B} par la rotation avec le barycentre du nuage de \mathcal{A} . La valeur de ε^2 dans ce cas est alors ε_L (L pour linéaire) :

$$\varepsilon_L^2(R) = \mathcal{S} \left[\|(a_i - \mathcal{S}[a_i]) - R(b_i - \mathcal{S}[b_i])\|^2 \right]. \quad (4.5)$$

On voit qu'il est judicieux de considérer les coordonnées des deux nuages par rapport à leur barycentre pour simplifier l'écriture des équations. On définit ainsi $a'_i = a_i - \mathcal{S}[a_i]$ et $b'_i = b_i - \mathcal{S}[b_i]$. En utilisant ces notations ε_L est défini par :

$$\varepsilon_L^2(R) = \mathcal{S} \left[\|a'_i - R(b'_i)\|^2 \right]. \quad (4.6)$$

La résolution du problème 1 passe donc par la résolution du problème linéaire associé dans lequel l'inconnue est la rotation de la transformation rigide φ (sa partie linéaire). Ce problème s'énonce :

Problème 2 (Problème linéaire associé)

Etant données deux familles de vecteurs a_i et b_i ($i = 1 \dots n$) trouver une rotation $R \in SO_d$ minimisant ε_L défini par la relation 4.6.

Une fois trouvée la rotation R solution de ce problème la translation de vecteur $\mathcal{S}[a_i] - R(\mathcal{S}[b_i])$ permet d'obtenir la transformation rigide solution du problème 1.

Remarquons que la relation 4.6 se développe en :

$$\begin{aligned} \varepsilon_L^2(R) &= \mathcal{S} \left[\|a'_i\|^2 \right] + \mathcal{S} \left[\|b'_i\|^2 \right] - 2\mathcal{S} \left[\langle a'_i, R(b'_i) \rangle \right] \\ &= \left(\mathcal{S} \left[\|a_i\|^2 \right] - \|\mathcal{S}[a_i]\|^2 \right) + \left(\mathcal{S} \left[\|b_i\|^2 \right] - \|\mathcal{S}[b_i]\|^2 \right) \\ &\quad - 2\mathcal{S} \left[\langle a_i - \mathcal{S}[a_i], R(b_i - \mathcal{S}[b_i]) \rangle \right]. \end{aligned} \quad (4.7)$$

On voit que seul le dernier terme dépend de R et qu'il doit être maximal pour minimiser ε_L . Le problème 2 est donc équivalent au problème suivant :

Problème 3 (Problème linéaire réduit)

Etant données deux familles de vecteurs a_i et b_i ($i = 1 \dots n$) trouver une rotation $R \in SO_d$ maximisant le "coefficient de corrélation" ω défini par :

$$\omega(R) = \mathcal{S} [\langle a'_i, R(b'_i) \rangle] \quad (4.8)$$

Intuitivement ce coefficient est maximal quand pour chaque point i l'image de b'_i par la rotation coïncide avec a'_i .

Les solutions du problème de mise en correspondance de nuages de points appariés proposées dans la littérature diffèrent par la représentation de la rotation inconnue et le point de départ des dérivations : minimiser ε_L ou maximiser ω . Les différentes représentations des rotations sont abordées en détail dans le chapitre 2 de cette partie. Nous ne reviendrons donc pas ici sur la définition et les propriétés de ces représentations. Nous voyons dans les sections suivantes les solutions apportées aux problèmes 2 et 3 dans la littérature.

4.2.3 Grandeurs associées au problème

Afin d'augmenter la clarté de cet exposé nous définissons ici différentes grandeurs liées au problème qui nous seront utiles par la suite.

Nous allons utiliser les matrices d'inertie des deux nuages de points que nous noterons A et B :

$$\begin{aligned} A &= \mathcal{S} [a'_i a_i'^T] = \mathcal{S} [a_i a_i^T] - \mathcal{S} [a_i] \mathcal{S} [a_i]^T, \\ B &= \mathcal{S} [b'_i b_i'^T] = \mathcal{S} [b_i b_i^T] - \mathcal{S} [b_i] \mathcal{S} [b_i]^T. \end{aligned} \quad (4.9)$$

La matrice d'inertie "croisée" appelée matrice de corrélation ou encore matrice de covariance croisée notée Ψ est définie de façon analogue par :

$$\Psi = \mathcal{S} [a'_i b_i'^T] = \mathcal{S} [a_i b_i^T] - \mathcal{S} [a_i] \mathcal{S} [b_i]^T. \quad (4.10)$$

Notons que si on échange les deux nuages la matrice de corrélation du nouveau problème est la transposée de celle-ci.

Les parties symétrique et antisymétrique de Ψ jouent aussi un rôle important nous noterons Λ la partie symétrique de Ψ :

$$\Lambda = \frac{\Psi + \Psi^T}{2}. \quad (4.11)$$

Dans le cas de la dimension 3 nous noterons Γ le vecteur représentant la partie antisymétrique de Ψ (qui ne dépend en effet dans ce cas que de trois coefficients) :

$$\Gamma = \begin{pmatrix} \Psi_{32} - \Psi_{23} \\ \Psi_{13} - \Psi_{31} \\ \Psi_{21} - \Psi_{12} \end{pmatrix}. \quad (4.12)$$

4.2.4 Solutions utilisant la représentation matricielle

On note Q la matrice de la rotation inconnue R . On remarque que le coefficient de corrélation ω s'écrit en fonction de la matrice de corrélation Ψ :

$$\omega(R) = \text{Tr } Q^T \Psi. \quad (4.13)$$

La trace de $Q^T \Psi$ étant simplement la somme $\sum Q_{ij} \Psi_{ij}$.

Cette écriture conduit immédiatement à la solution dans le cas $d = 2$:

$$Q = \frac{\begin{pmatrix} \Psi_{11} + \Psi_{22} & \Psi_{12} - \Psi_{21} \\ \Psi_{21} - \Psi_{12} & \Psi_{11} + \Psi_{22} \end{pmatrix}}{\sqrt{(\Psi_{11} + \Psi_{22})^2 + (\Psi_{12} - \Psi_{21})^2}}. \quad (4.14)$$

Ce résultat en dimension 2 est ébauché par Lin *et al.* dans [78] et donné explicitement par Haralick *et al.* dans [60]. Dans la suite de cette section on considère uniquement le cas $d = 3$.

En dimension 3 des travaux existent depuis 1936 (voir l'article de Horn *et al.* [65] pour les références). Horn, Hilden, Negahdaripour [65] et Arun, Huang, Blostein [2] ont proposé vers 1987 deux approches différentes utilisant la représentation matricielle des rotations. L'approche de Horn *et al.* est basée sur la décomposition polaire de la matrice Ψ obtenue en diagonalisant la matrice symétrique $\Psi^T \Psi$; celle de Arun *et al.* utilise la décomposition en valeurs singulières de Ψ .

Umeyama [109] propose une correction à l'algorithme de Arun *et al.* qui pouvait donner une matrice orthogonale qui ne soit pas une rotation.

Kanatani [70] simplifie la démonstration de Umeyama et résume les différentes solutions du problème qui compris les solutions basées sur les quaternions.

Solution de Horn *et al.*

On montre [85] qu'il existe une matrice orthogonale $O \in O_d$ et une unique matrice symétrique positive S (i.e. $S^T = S$ et $\forall u \in \mathbb{R}^d, \langle Su, u \rangle \geq 0$) telles que $\Psi = OS$ (factorisation polaire). O est déterminée de manière unique sur l'image de S et est donc unique quand S est inversible c'est à dire quand Ψ l'est.

Alors si $\det O = 1$ (cas le plus fréquent) $Q = O$ est une solution du problème sinon $Q = O(I - 2uu^T)$ est une solution du problème avec u un vecteur propre unitaire associé à la plus petite valeur propre de S (voir [70] pour une démonstration).

Dans leur article Horn *et al.* proposent de calculer S en diagonalisant la matrice $\Psi^T \Psi$ on a alors :

$$O = \Psi(\Psi^T \Psi)^{-\frac{1}{2}}. \quad (4.15)$$

Dans le cas où Ψ n'est pas inversible la pseudo inverse de la racine carrée fournit une solution.

Solution de Arun *et al.*

On montre qu'il existe deux matrices orthogonales $U, V \in O_d$ et une matrice diagonale $D = \text{Diag}(\lambda_1, \lambda_2, \lambda_3)$ avec $\lambda_1 \geq \lambda_2 \geq \lambda_3$ telles que $\Psi = VDU^T$ (décomposition en valeurs singulières).

Alors $\Gamma Q = V \text{Diag}(1, 1, \det VU^T)U^T$ est une solution du problème. Le cas le plus fréquent correspond à $\det VU^T = 1$ (voir [70] pour une démonstration).

Le calcul de la décomposition en valeurs singulières (*SVD*) se trouve dans la plupart des bibliothèques de fonctions d'algèbre linéaire (*lapack* Γ *numerical recipes* Γ ..).

Lien entre les deux solutions

Le calcul de la racine carrée positive de la matrice symétrique positive $\Psi^T \Psi \Gamma$ servant à calculer la factorisation polaire de la matrice Ψ s'effectue par diagonalisation dans une base orthogonale :

$$\Psi^T \Psi = \Omega \Delta \Omega^T. \quad (4.16)$$

Ω est une matrice orthogonale dont les colonnes sont les vecteurs propres de $\Psi^T \Psi$ et Δ est une matrice diagonale dont les éléments sont les valeurs propres de $\Psi^T \Psi$.

La factorisation polaire de Ψ se déduit de cette diagonalisation :

$$\begin{aligned} \Psi &= O S, \\ O &= \Psi \Omega \Delta^{-\frac{1}{2}} \Omega^T, \\ S &= \Omega \Delta^{\frac{1}{2}} \Omega^T. \end{aligned} \quad (4.17)$$

On remarque que la décomposition en valeurs singulières de Ψ peut être extraite trivialement de ces relations :

$$\begin{aligned} \Psi &= U D V^T, \\ U &= O \Omega, \\ D &= \Delta^{\frac{1}{2}}, \\ V &= \Omega. \end{aligned} \quad (4.18)$$

Dans l'autre sens Γ si on connaît la décomposition en valeurs singulières de $\Psi \Gamma$ on peut en déduire simplement sa factorisation polaire :

$$\begin{aligned} O &= U V^T, \\ S &= V D V^T. \end{aligned} \quad (4.19)$$

4.2.5 Solutions utilisant les quaternions unitaires

Il existe deux méthodes dans la littérature.

La première a été présentée en 1986 par Faugeras et Hébert [47] [46]. On la retrouve expliquée plus en détail dans le livre de Horaud et Monga [63]. Cette méthode prend le problème linéaire de minimisation (problème 2) comme point de départ pour introduire la représentation de R par un quaternion unitaire.

La seconde est due à Horn en 1987 [64]. Elle maximise le critère de corrélation (problème 3) en y introduisant les quaternions.

Solution de Faugeras et Hébert

Si q est un quaternion unitaire associé à R la relation 4.6 s'écrit :

$$\varepsilon_L^2(R) = \mathcal{S} \left[\|a'_i - q * b'_i * \bar{q}\|^2 \right], \quad (4.20)$$

soit encore en multipliant à droite par q de norme 1 :

$$\varepsilon_L^2(R) = \mathcal{S} \left[\|a'_i * q - q * b'_i\|^2 \right]. \quad (4.21)$$

Le produit à droite ou à gauche par un quaternion est linéaire et peut donc se mettre sous la forme du produit à gauche par une matrice carrée de dimension 4. On se référera au chapitre 2 de cette partie section 2.6 pour plus de détails à ce sujet et pour la définition de Q_g et Q_d . On a alors :

$$a'_i * q = Q_g(a'_i) q, \quad (4.22)$$

$$q * b'_i = Q_d(b'_i)^T q. \quad (4.23)$$

En utilisant ces matrices ε_L s'écrit :

$$\begin{aligned} \varepsilon_L^2(R) &= \mathcal{S} \left[\|(Q_g(a'_i) - Q_d(b'_i))q\|^2 \right] \\ &= q^T \Lambda_\varepsilon q. \end{aligned} \quad (4.24)$$

en définissant la matrice symétrique positive (par construction) :

$$\Lambda_\varepsilon = \mathcal{S} \left[(Q_g(a'_i) - Q_d(b'_i))^T (Q_g(a'_i) - Q_d(b'_i)) \right]. \quad (4.25)$$

Le quaternion q recherché est alors un vecteur propre unitaire associé à la plus petite valeur propre de Λ_ε . Dans le cas usuel l'espace propre associé à cette valeur propre est de dimension 1 et on a le choix entre deux quaternions opposés qui définissent bien la même rotation (voir le chapitre précité).

Solution de Horn

Horn introduit la représentation de la rotation inconnue par un quaternion unitaire dans la définition du coefficient de corrélation ω . La dérivation des équations s'effectue selon des étapes similaires à celles de la méthode de Faugeras et Hébert :

$$\begin{aligned} \omega(R) &= \mathcal{S} [\langle a'_i, q * b'_i * \bar{q} \rangle] \\ &= \mathcal{S} [\langle a'_i * q, q * b'_i \rangle] \\ &= \mathcal{S} [\langle Q_g(a'_i) q, Q_d(b'_i)^T q \rangle] \\ &= q^T \Lambda_\omega q, \end{aligned} \quad (4.26)$$

Λ_ω étant la matrice symétrique :

$$\Lambda_\omega = \mathcal{S} \left[Q_g(a'_i)^T Q_d(b'_i)^T \right]. \quad (4.27)$$

Le quaternion q recherché est dans ce cas un vecteur propre unitaire associé à la plus grande valeur propre de Λ_ω . Notons qu'il ne s'agit pas de la plus grande valeur propre en module (les valeurs propres peuvent en effet être négatives).

Horn calcule explicitement les coefficients de Λ_ω . On peut les exprimer en fonction de ceux de la matrice de corrélation Ψ définie dans la section précédente (la différence des signes par rapport à la matrice donnée dans [64] est due au fait que dans l'article de Horn Γ c'est le nuage \mathcal{A} qui subit la transformation Γ et non pas \mathcal{B} comme dans notre exposé) :

$$\Lambda_\omega = \begin{pmatrix} \text{Tr } \Psi & \Psi_{32} - \Psi_{23} & \Psi_{13} - \Psi_{31} & \Psi_{21} - \Psi_{12} \\ \Psi_{32} - \Psi_{23} & 2\Psi_{11} - \text{Tr } \Psi & \Psi_{12} + \Psi_{21} & \Psi_{13} + \Psi_{31} \\ \Psi_{13} - \Psi_{31} & \Psi_{12} + \Psi_{21} & 2\Psi_{22} - \text{Tr } \Psi & \Psi_{23} + \Psi_{32} \\ \Psi_{21} - \Psi_{12} & \Psi_{13} + \Psi_{31} & \Psi_{23} + \Psi_{32} & 2\Psi_{33} - \text{Tr } \Psi \end{pmatrix}. \quad (4.28)$$

La matrice Λ_ω s'écrit par blocs en fonction des grandeurs définies plus haut :

$$\Lambda_\omega = \left(\begin{array}{c|c} \text{Tr } \Psi & \Gamma^T \\ \hline \Gamma & 2\Lambda - (\text{Tr } \Psi)I_3 \end{array} \right). \quad (4.29)$$

Lien entre les deux solutions

D'après la relation 4.7 l'erreur ε_L^2 est la différence entre des grandeurs ne dépendant que de la forme des deux nuages de points Γ indépendamment de la rotation $R\Gamma$ et le coefficient de corrélation ω .

De même il s'avère que la matrice Λ_ε est la différence entre une matrice ne dépendant que de la forme des nuages et la matrice Λ_ω ; plus précisément :

$$\Lambda_\varepsilon = (\mathcal{S} [a_i^2] + \mathcal{S} [b_i^2])I_4 - 2\Lambda_\omega. \quad (4.30)$$

La recherche de la plus petite valeur propre de Λ_ε est donc identique à celle de la plus grande valeur propre de Λ_ω .

4.2.6 Solutions utilisant le vecteur rotation

Horaud et Monga [63] proposent une approche différente du problème de mise en correspondance de nuages de points appariés. Cette approche adopte une autre définition de la distance entre nuages et conduit à des équations plus simples.

Nous proposons dans cette section une autre méthode Γ minimisant l'erreur aux moindres carrés utilisée dans les sections précédentes.

Ceci permettra d'établir que la solution apportée par la méthode de Horaud et Monga est aussi solution du problème 2 dans le cas idéal où l'erreur aux moindres carrés est nulle.

Dans ces deux paragraphes Γk est un vecteur directeur unitaire de l'axe de la rotation R et θ son angle.

Solution de Horaud et Monga

Dans [63] des considérations géométriques sur la mise en correspondance des couples de points conduisent les auteurs à une relation qui doit être idéalement vérifiée par chaque couple de points :

$$(a'_i + b'_i) \wedge K = (b'_i - a'_i), \quad (4.31)$$

en posant $K = (\tan \frac{\theta}{2})k$.

On sait d'autre part qu'un produit vectoriel peut se mettre sous forme d'un produit par une matrice antisymétrique :

$$u \wedge K = \tilde{u}K = \begin{pmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{pmatrix} K. \quad (4.32)$$

L'écriture de l'égalité 4.31 pour les n couples de points fournit un système surcontraint de $3n$ équations à 3 inconnues que l'on résout classiquement par la méthode des moindres carrés. Notons que les 3 équations fournies par un couple de points ne sont pas indépendantes car les matrices antisymétriques sont singulières. Le système a ainsi au plus $2n$ équations indépendantes.

On cherche donc le vecteur K minimisant :

$$\mathcal{S} \left[\|H(a'_i + b'_i)K - (b'_i - a'_i)\|^2 \right], \quad (4.33)$$

c'est à dire la solution du système :

$$\mathcal{S} \left[H(a'_i + b'_i)^T H(a'_i + b'_i) \right] K = \mathcal{S} \left[H(a'_i + b'_i)^T (b'_i - a'_i) \right]. \quad (4.34)$$

Ce système peut s'écrire en fonction de la matrice Ψ des deux matrices d'inertie des nuages A et B et du vecteur Γ :

$$((\text{Tr } A + \text{Tr } B + 2 \text{Tr } \Psi)I_3 - (A + B + 2\Lambda))K = 2\Gamma. \quad (4.35)$$

Lorsque l'erreur de mise en correspondance est nulle (les deux nuages sont égaux à une transformation rigide près) la construction géométrique de la méthode permet d'affirmer que le résultat obtenu est identique à celui des autres méthodes et correspond bien à la rotation recherchée.

Nous avons constaté expérimentalement que le résultat de cette méthode dans le cas où l'erreur n'est pas nulle était différent de celui fourni par les autres méthodes.

Une autre solution

k et θ représentant toujours l'axe (unitaire) et l'angle de R nous avons vu au chapitre 2 de cette partie que $Q = I_3 + \sin \theta \tilde{k} + (1 - \cos \theta)k^2$. En définissant la matrice Λ_k par :

$$\Lambda_k = \Lambda - (\text{Tr } \Psi)I_3, \quad (4.36)$$

le coefficient de corrélation ω s'écrit :

$$\omega(R) = \text{Tr } \Psi - \sin \theta \langle k, \Gamma \rangle + (1 - \cos \theta) \langle k, \Lambda_k k \rangle. \quad (4.37)$$

Pour un vecteur unitaire $k \in S_2$ donné Γ on peut calculer explicitement la valeur $\theta_0(k)$ de θ rendant ω maximal avec :

$$\begin{aligned} \sin(\theta_0(k)) &= \frac{-\langle k, \Gamma \rangle}{\sqrt{\langle k, \Lambda_k k \rangle^2 + \langle k, \Gamma \rangle^2}}, \\ \cos(\theta_0(k)) &= \frac{-\langle k, \Lambda_k k \rangle}{\sqrt{\langle k, \Lambda_k k \rangle^2 + \langle k, \Gamma \rangle^2}}. \end{aligned} \quad (4.38)$$

Pour k fixé Γ la valeur maximale que peut prendre ω est donc :

$$\omega_0(k) = \text{Tr } \Psi + \langle k, \Lambda_k k \rangle + \sqrt{\langle k, \Lambda_k k \rangle^2 + \langle k, \Gamma \rangle^2}. \quad (4.39)$$

Le gradient de ω_0 selon k (sur $\mathbb{R}^3 \Gamma$ sans tenir compte de la contrainte $\|k\| = 1$) est :

$$\text{grad } \omega_0(k) = 2\Lambda_k k + \frac{\langle k, \Gamma \rangle \Gamma + \langle k, \Lambda_k k \rangle \Lambda_k k}{\sqrt{\langle k, \Lambda_k k \rangle^2 + \langle k, \Gamma \rangle^2}}. \quad (4.40)$$

Quand ω_0 est maximal pour $\|k\| = 1$ il est nécessaire que les deux vecteurs k et $\text{grad } \omega_0(k)$ soient colinéaires.

On peut donc envisager d'itérer $n_{k+1} = \text{grad } \omega_0(n_k)$ en normalisant n_{k+1} après chaque itération. Si les itérations convergent Γ on aura alors une solution.

Nous avons vérifié expérimentalement que cette méthode converge dans la plupart des cas Γ et le temps de calcul nécessaire (à précision du résultat égale) est la moitié de celui nécessité par la méthode de Arun *et al.* Il subsiste néanmoins des cas dans lesquels les itérations ne convergent pas.

4.3 Conclusion

Après avoir défini le problème général de la mise en correspondance Γ nous avons étudié le cas particulier de la mise en correspondance de nuages de points appariés Γ en minimisant un critère de moindres carrés.

Nous avons montré comment ce problème se ramène à la recherche d'une rotation maximisant un "critère de corrélation". Nous rassemblons et comparons ensuite les différentes méthodes proposées dans la littérature pour résoudre ce problème. Nous proposons finalement une nouvelle solution itérative Γ utilisant la représentation des rotations par leur "vecteur rotation".

La résolution efficace de ce problème présente un intérêt industriel important. En effet Γ ce calcul est effectué en permanence dans les systèmes de localisation tridimensionnelle Γ tels que l'Optotrak (R).

Certaines voies restent encore à explorer dans ce problème Γ qui permettront sans doute d'améliorer encore les algorithmes :

- La méthode que nous avons présentée est itérative Γ et ne converge pas dans certains cas. Etant donné qu'elle est plus rapide que la méthode de Arun *et*

al. dans les cas où elle converge il est intéressant de l'étudier plus en profondeur. On peut aussi chercher une solution explicite en partant de la dernière équation.

- On constate que la factorisation polaire $\Psi = OS$ de la matrice Ψ peut être déduite de celle de $Q\Psi$ pour toute matrice de rotation Q ou de celle de ΨT pour toute matrice symétrique T . On peut alors se demander s'il n'est pas possible de tirer parti de ces remarques pour ramener Ψ à une forme dont la factorisation polaire est facile à calculer.
- Enfin quand les deux nuages correspondent exactement on a la propriété $\Psi = AQ = QB$. Les matrices $M_1 = A^{-1}\Psi B^{-1}\Psi^T$ et $M_2 = B^{-1}\Psi^T A^{-1}\Psi$ sont dans ce cas l'identité. On peut se demander s'il y a un lien direct entre l'erreur minimale et les coefficients de ces matrices.

Dans un autre ordre d'idées on peut se poser la question "Quelle est la meilleure forme à donner à un nuage de points?". Par exemple dans le cas où l'on doit positionner n émetteurs ponctuels sur un objet pour le repérer ensuite avec un localisateur tridimensionnel.

Chapitre 5

Segmentation

La segmentation est une des opérations de l'analyse d'images. Elle consiste à partir d'une image à la partitionner de telle sorte que les régions ainsi définies correspondent aux objets présents dans la scène dont l'image est issue.

Aborder le problème général de la segmentation sortirait largement du contexte de cet exposé. Nous renvoyons le lecteur à l'ouvrage collectif coordonné par Cocque-
rez et Philipp [20] qui fournit un panorama très complet des différentes approches proposées pour résoudre ce problème.

Nous allons traiter un cas particulier de segmentation dans lequel on connaît précisément la géométrie de certains objets de la scène. La segmentation consiste alors à localiser dans l'image les régions correspondant à ces objets.

Nous avons été confrontés à ce problème lors de la recherche de la position des centres de petites sphères métalliques dans une image scanner (cf partie *Simulation et assistance d'un geste chirurgical*). Ce problème apparaît aussi dans le domaine de la vision par ordinateur dans lequel des objets à la géométrie connue (grilles, sphères, bandes...) servent au calibrage des caméras.

Nous nous intéressons ici au cas des objets circulaires : cercles, disques et motifs d'anneaux concentriques. Les objets circulaires sont fréquemment utilisés en vision par ordinateur. Bose et Amir [22] ainsi que Efrat et Gotsman [44] montrent que ces objets permettent un repérage sub-pixel plus précis que d'autres formes simples avec en plus l'avantage de l'indépendance vis-à-vis de la rotation de la caméra.

Ce chapitre comporte deux sections.

Nous traitons en premier lieu (section 5.1) du problème de la détection grossière de toutes les projections d'objets circulaires ou elliptiques dans une image. Le résultat de cette détection est un ensemble de centres possibles pour les objets recherchés.

Ensuite nous exposons un travail original sur la **détection globale** d'une grille d'objets circulaires sur une image : plutôt que de chercher à localiser précisément chaque objet il est plus précis et efficace de rechercher la grille dans son intégralité.

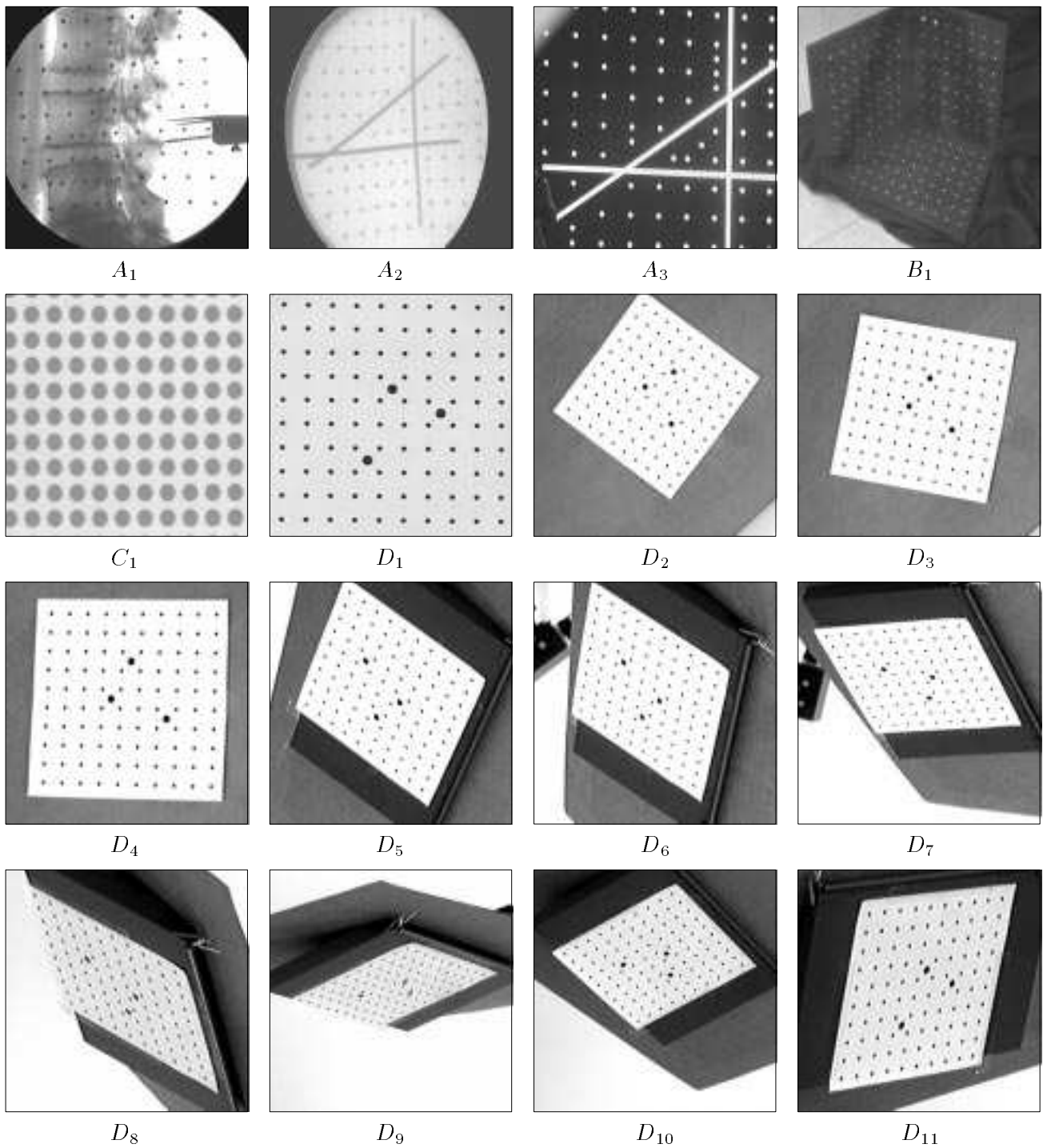


Fig. 43 - Nos 16 images de test; leur provenance est expliquée dans le texte.

La figure 43 montre 16 images de test réparties en quatre séries. Les images de la série *A* sont des radios et des vidéos de mires utilisées pour calibrer des amplificateurs de brillance et des caméras. Elles ont été fournies par P. Sautot et A. Hamadeh. L'image *B* est celle d'une mire de calibrage formée de trois plansΓfournie par P. Brand. L'image *C* est due à M. Puech ; il s'agit d'une plaquette de microscope sur laquelle sont fixées des pastilles d'épaisseur connue. Les 11 images vidéo de la série *D* sont obtenues à partir d'une même mire expérimentale (les images ont été acquises grâce à la collaboration d'A. Poyet). Cette mire est une grille régulière générée en PostScript ; on peut ainsi utiliser la précision des imprimantes laser pour réaliser des tests au moindre coût. Cette série permet de constater les effets de l'angle de projection sur le résultat.

Sur toutes ces mires servant à calibrer des systèmes d'acquisition d'imagesΓon utilise des objets circulaires dont les projections font quelques pixels de diamètre. Les images de la série *A* utilisent également des segmentsΓqui servent à orienter la mire.

5.1 Détection grossière d'éléments circulaires et elliptiques

Le problème de la détection grossière d'éléments circulaires peut se poser ainsi :

Problème 4 (Détection grossière d'éléments circulaires)

Etant donnée une image d'une scène comportant des disquesΓtrouver tous les centres et les rayons des projections de ces disques dans l'imageΓavec une précision de l'ordre du pixel.

Ce problème se pose de la même façon dans le cas d'images tridimensionnelles et d'éléments elliptiques en 2D ou "ellipsoïdaux" en 3D.

5.1.1 Approches existantes

La plupart des algorithmes existants sont des généralisations de la transformée de Hough.

La transformée de Hough permet la détection d'objets paramétrés dans une image.

Par exempleΓdans le cas où l'on recherche les cercles sur une imageΓles paramètres sont les coordonnées de son centre x et y Γainsi que son rayon r . On dispose d'une fonction de discrétisationΓqui fournit pour une valeur donnée des paramètres (x, y, r) l'ensemble des pixels de l'image appartenant au cercle correspondant.

On calcule alors la somme $s(x, y, r)$ des valeurs de l'image en ces points. Cette somme est maximale quand (x, y, r) correspond à un cercle effectivement présent sur l'image. Il suffit alors de rechercher les maxima de s quand (x, y, r) prend un ensemble fini de valeurs acceptables fixé au départΓpar exemple (x, y) dans l'image et $0 < r < 10$.

La transformée de Hough de l'image initiale est dans ce cas l'image de dimension 3 associant à chaque triplet (x, y, r) la valeur de s en ce point.

Initialement la transformée de Hough concernait la détection de segments. Dans ce cas le nombre de paramètres est réduit à 2 et permet d'effectuer la transformation en un temps raisonnable.

Illingworth et Kittler exposent dans [67] un résumé bibliographique sur la transformée de Hough. Dans [114] Yip et Tam et Leung résument les différentes extensions et améliorations de la transformée de Hough utilisées pour détecter des cercles et des ellipses. D'autres références sur ce sujet sont données dans [74] et [71].

Des propriétés géométriques des cercles et des ellipses sont utilisées par Ho et Chen [62] pour réduire le problème à la recherche de segments dans l'image effectuée par transformée de Hough classique.

Le principal problème de la transformée de Hough et de ses extensions est le temps de calcul qui rend l'algorithme inutilisable dès que le nombre de paramètres dépasse 3. C'est pourquoi certains auteurs ont abordé la parallélisation de l'algorithme. Kumar, Ranganathan et Goldgof [72] proposent des algorithmes parallèles pour la détection de cercles ainsi qu'une bibliographie sur les parallélisations de la transformée de Hough.

5.1.2 Solution proposée

L'algorithme que nous proposons permet de détecter en une seule passe l'ensemble des objets dont le rayon est compris entre deux bornes données r_{min} et r_{max} (il s'agit du rayon des projections des objets sur l'image et non pas du rayon des objets réels).

Cet algorithme est basé sur l'hypothèse que la région de l'image correspondant à l'intérieur de l'objet est homogène ainsi que la région extérieure entourant immédiatement l'objet. Cette hypothèse est généralement bien vérifiée dans les images que nous avons traitées. Même dans le cas des mires "radio" (série A) où le fond n'est pas homogène seuls quelques points se trouvent à cheval sur des zones d'intensités différentes.

Critère de présence d'un objet circulaire

L'idée est donc simple : pour chaque point $m = (x, y)$ de l'image on calcule la moyenne et l'écart-type des valeurs des pixels dans un disque discret de rayon r_{min} centré en m et dans une couronne discrète de rayon interne r_{max} de même centre. A partir de ces 4 grandeurs on calcule un coefficient $\chi(m)$ mesurant la "probabilité de présence" d'un objet circulaire centré en m .

Formalisons ces définitions :

Definition 23 (Région)

On appelle région une partie finie de \mathbb{Z}^2 . Si A est une région et $m \in \mathbb{Z}^2$ un vecteur on définit la région $m + A$ par $m + A = \{m + x, x \in A\}$. Le cardinal d'une région A sera noté $|A|$.

Definition 24 (Image)

Nous appellerons image sur p bits de taille $n_x \times n_y$ une fonction à valeurs dans l'ensemble $\{0, 1, \dots, 2^p - 1\}$ et définie sur $\{1, 2, \dots, n_x\} \times \{1, 2, \dots, n_y\}$; cette région sera appelée le *support* de l'image. Les images rencontrées usuellement sont sur 1 bit

(images binaires) sur 8 bits (images en niveaux de gris vidéo) et 12 bits (images en niveaux de gris médicales : TDMIRM).

Definition 25 (Moyenne et écart-type d'une image sur une région)

Soit F une image de support S et A une région incluse dans S . On définit la moyenne des valeurs de F sur A notée $\mu_F(A)$ et l'écart-type des valeurs de F sur A noté $\sigma_F(A)$ par :

$$\begin{aligned}\mu_F(A) &= \frac{1}{|A|} \sum_{x \in A} F(x), \\ \sigma_F^2(A) &= \frac{1}{|A|} \sum_{x \in A} [F(x) - \mu_F(A)]^2, \\ &= \left[\frac{1}{|A|} \sum_{x \in A} F(x)^2 \right] - \mu_F(A)^2.\end{aligned}\tag{5.1}$$

On étend ces définitions au cas où A a simplement une intersection non vide avec S par : $\mu_F(A) = \mu_F(A \cap S)$ et de même pour σ . En l'absence d'ambiguïté on omettra l'indice F .

On note e l'épaisseur de la couronne extérieure E l'image et S son support. On définit alors deux régions : I la région intérieure et E la région extérieure par

$$\begin{aligned}I &= \{x, x \in \mathbb{Z}^2, \|x\| \leq r_{min}\}, \\ E &= \{x, x \in \mathbb{Z}^2, r_{max} \leq \|x\| \leq r_{max} + e\}.\end{aligned}\tag{5.2}$$

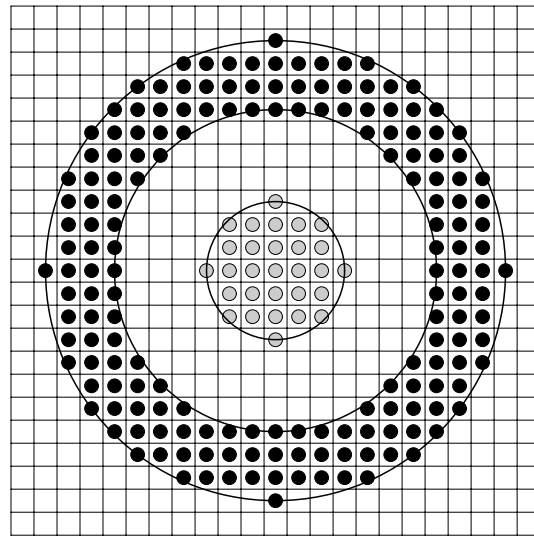


Fig. 44 - Régions intérieure I (en gris) et extérieure E (en noir) intervenant dans le calcul de χ . Dans cette figure, des cercles matérialisent le disque central et la couronne extérieure. On a pris $r_{min} = 3$, $r_{max} = 7$, et $e = 3$.

Intuitivement le terme $\chi(m)$ doit augmenter quand les écarts-types diminuent avec une prépondérance du terme intérieur. $\chi(m)$ doit diminuer quand la différence

des moyennes intérieure et extérieure diminue c'est à dire quand le contraste diminue. Une définition naturelle de χ prenant en compte ces considérations est :

$$\chi(m) = \frac{|\mu(m+I) - \mu(m+E)|}{\sigma(m+I) + 0.5\sigma(m+E)}. \quad (5.3)$$

Le cas où le dénominateur est nul peut se produire. Dans ce cas quand le numérateur est nul lui aussi on prendra $\chi(m) = 0$ (en effet l'image est alors uniforme autour du point m) et $\chi(m) = +\infty$ sinon.

Sélection des candidats

La sélection des candidats c'est à dire de l'ensemble des points de l'image susceptibles d'être centres d'objets circulaires s'effectue à partir de l'ensemble des valeurs de χ calculées en chaque point de l'image.

Si μ désigne la moyenne des valeurs de χ sur l'image et σ leur écart-type (on exclut les points de valeur infinie du calcul de ces grandeurs) nous avons expérimentalement constaté que le critère $\chi > \mu + 5\sigma$ donnait de bons résultats. On impose de plus aux candidats d'être des maxima locaux de χ dans un rayon r_{min} .

L'ensemble des candidats finalement retenus est donc constitué des points m de S vérifiant les conditions :

$$\begin{cases} \chi(m) > \mu + 5\sigma, \\ \forall m' \in S, \|m - m'\| < r_{min} \Rightarrow \chi(m') \leq \chi(m). \end{cases} \quad (5.4)$$

Résultats

La figure 45 montre les résultats obtenus pour les images de test. Les valeurs du triplet (r_{min}, r_{max}, e) sont indiquées en dessous de chaque figure. La détermination de ces valeurs se fait pour le moment par tâtonnements mais on peut envisager de la calculer à partir de l'image. Dans les images testées moins de 3 essais ont été nécessaires : il suffit de grossir un pixel (avec un logiciel de visualisation) et d'évaluer approximativement son rayon pour trouver des valeurs convenables.

Les tableaux suivants résument les résultats obtenus sur les images de test. On distingue deux types d'erreurs : les candidats qui ne correspondent pas à des marques de l'image et inversement les marques de l'image qui ne sont associées à aucun candidat.

Image	A_1	A_2	A_3	B_1	C_1	D_1	D_2	D_3
Marqueurs image	78	86	79	160	99	100	100	100
Candidats	83	88	78	216	99	100	116	107
Candidats faux	5	4	0	56	0	0	16	7
Marqueurs non détectés	0	2	1	0	0	0	0	0
Image	D_4	D_5	D_6	D_7	D_8	D_9	D_{10}	D_{11}
Marqueurs image	100	100	100	100	100	100	100	100
Candidats	104	114	103	122	34	106	114	115
Candidats faux	4	14	21	29	19	35	14	15
Marqueurs non détectés	0	0	18	7	85	29	0	0

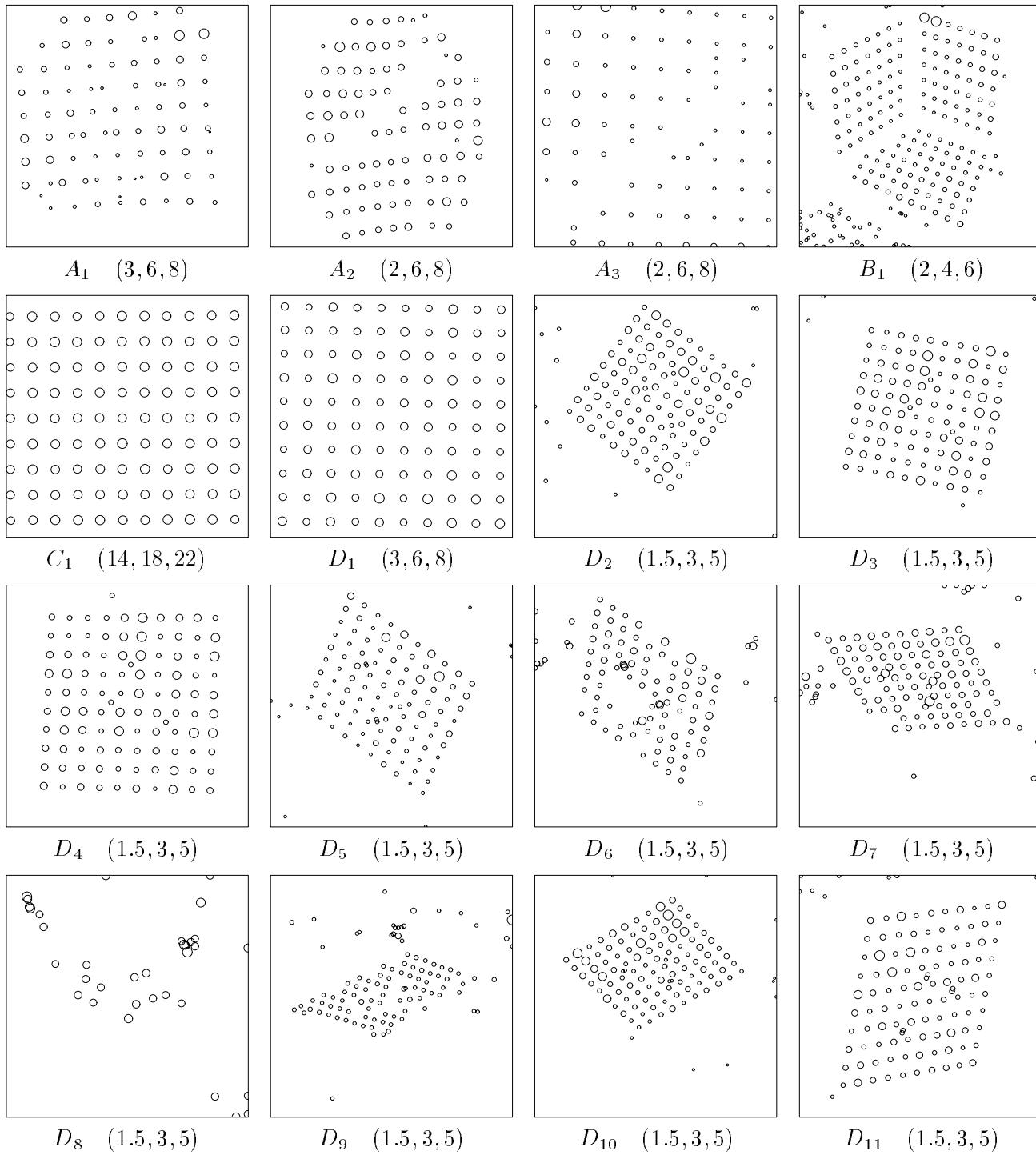


Fig. 45 - Centres détectés sur nos 16 images de test. Les paramètres employés pour chaque image sont indiqués sous la forme d'un triplet (r_{min}, r_{max}, e) . Le rayon des cercles est proportionnel à la valeur de χ (les échelles sont différentes d'une image à l'autre).

5.2 Localisation de la projection d'une grille régulière d'éléments circulaires

Dans cette section on cherche la position dans une image d'une grille régulière de disques. Les images des séries C et D contiennent de telles grilles.

Connaissant les dimensions réelles de la grille (le nombre de disques et la distance entre deux disques dans chaque direction) on souhaite retrouver la position de son projeté sur l'image.

Nous proposons ici quelques expériences menées dans ce sens. On note A_i pour $i = 1 \dots n$ l'ensemble des candidats et χ_i la valeur de χ au point A_i de l'image.

5.2.1 Signature

La première idée venant à l'esprit est de rechercher les directions prépondérantes d'alignement des candidats. Pour ce faire nous proposons pour deux candidats A_i et A_j distincts de calculer d'une part l'angle θ_{ij} que fait la droite $A_i A_j$ avec l'horizontale (θ_{ij} est dans $[0, \pi[$) et d'autre part le scalaire $d_{ij} = \frac{\chi_i \chi_j}{\|A_i A_j\|}$.

On fait alors la somme des pics de hauteur d_{ij} et d'abscisse θ_{ij} :

$$s(\theta) = \sum_{i \neq j} d_{ij} \delta(\theta - \theta_{ij}). \quad (5.5)$$

Cette fonction est ensuite convoluée par une gaussienne échantillonnée régulièrement sur $[0, \pi[$ puis ses valeurs sont ramenées à l'intervalle $[0, 1]$. Les graphes des fonctions ainsi obtenues sont présentés figure 46.

Les angles des projetés des deux directions principales de la grille sont ensuite déterminés en recherchant le maximum absolu de ces fonctions puis le maximum local formant approximativement un angle droit avec cette première direction. La figure 47 montre les angles obtenus superposés aux images initiales. On voit que cette heuristique ne donne pas toujours les résultats escomptés (images D_6 , D_8 et D_9).

5.2.2 Corrélation angulaire

Une autre approche consiste à regarder dans quelles directions les projections des candidats sont regroupées. On fait donc tourner une droite passant par l'origine en projetant orthogonalement tous les candidats dessus. Pour certains angles les points se regroupent et on cherche les valeurs de l'angle formant des groupes de taille homogène régulièrement répartis sur la droite. On obtient ainsi non seulement les directions principales de la grille mais aussi la période et le décalage selon ces directions.

La figure 48 montre les résultats obtenus avec cette seconde méthode.

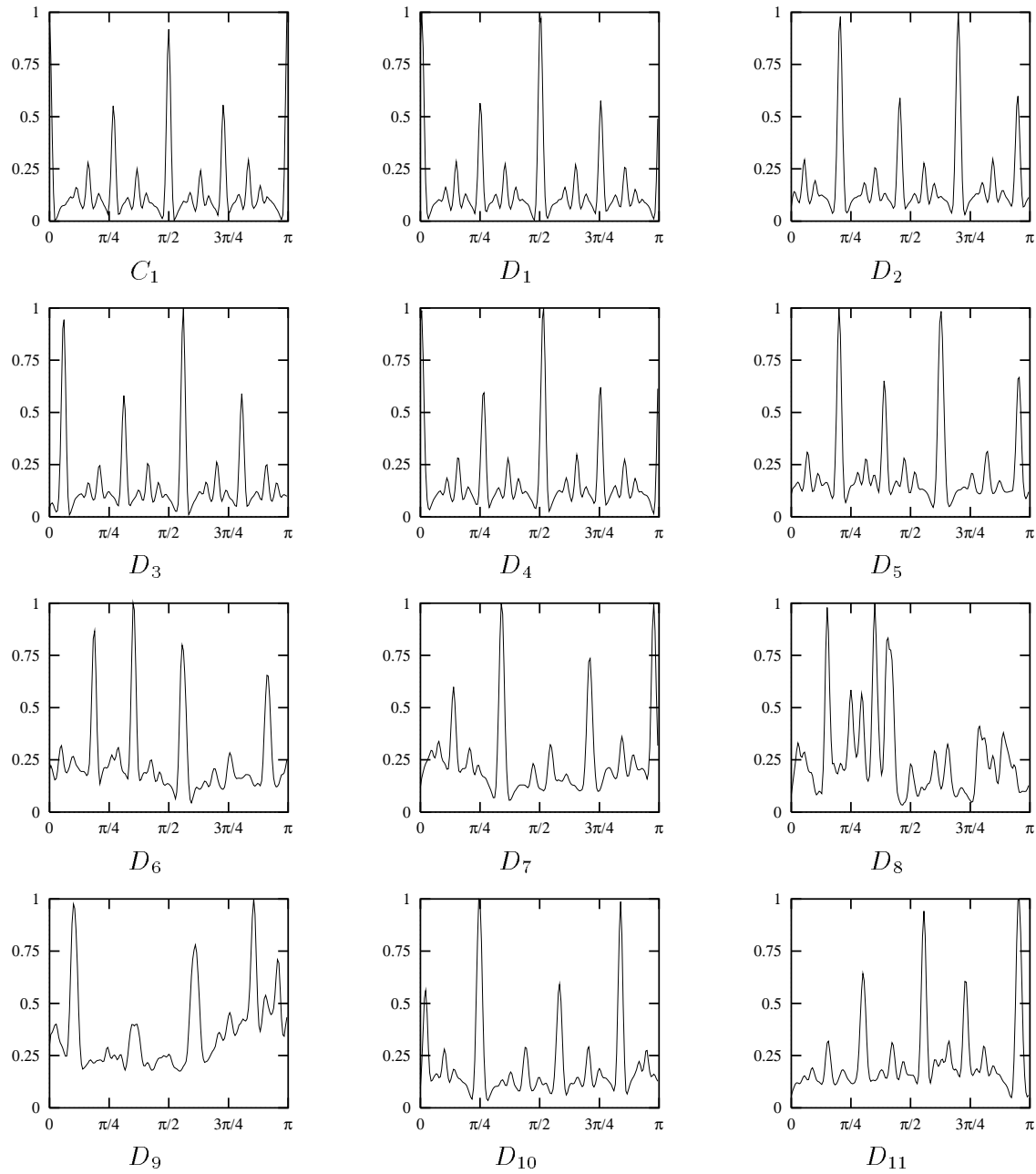


Fig. 46 - Signatures obtenues à partir des images des séries C et D .

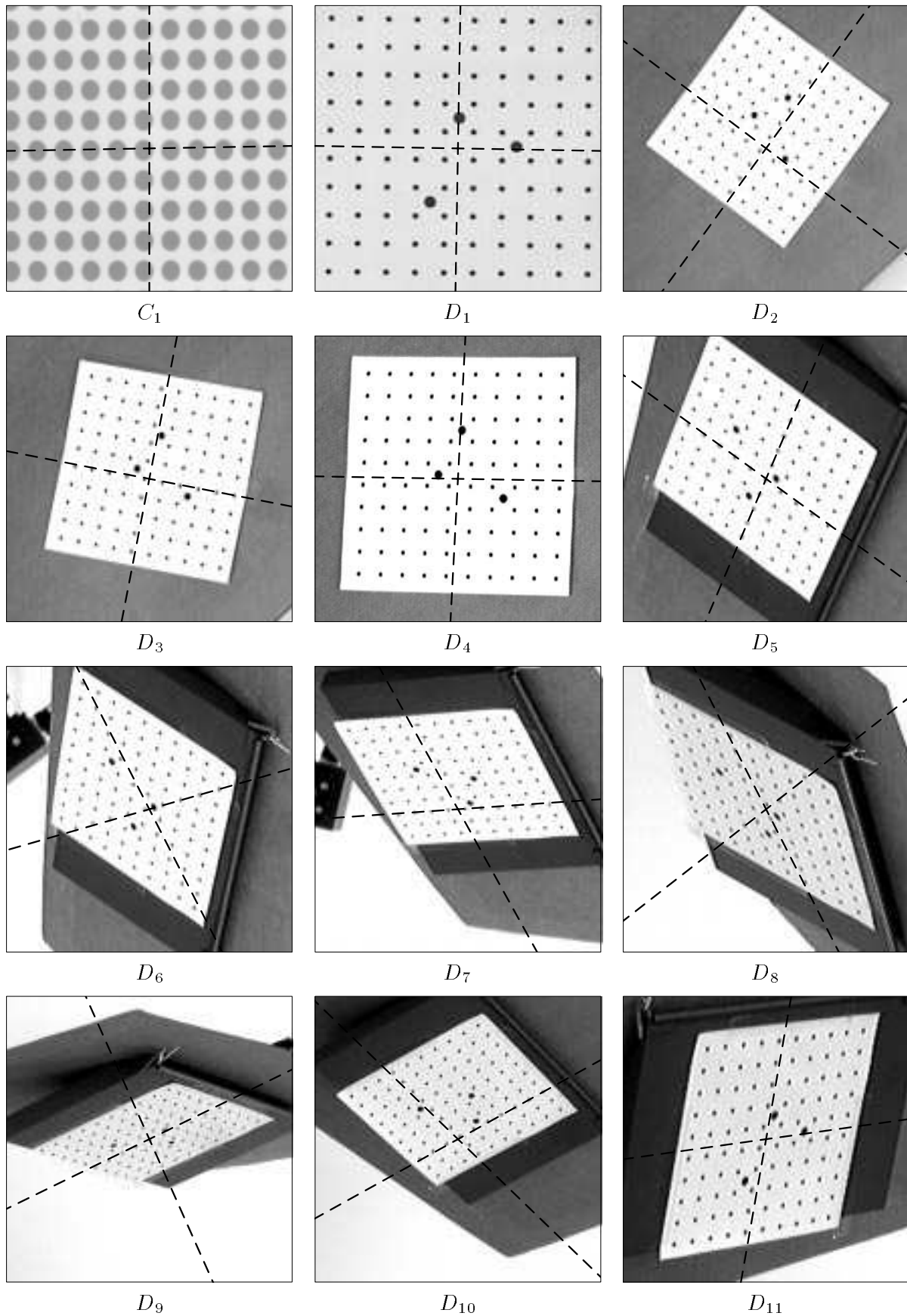


Fig. 47 - Angles déduits des signatures de la figure précédente.

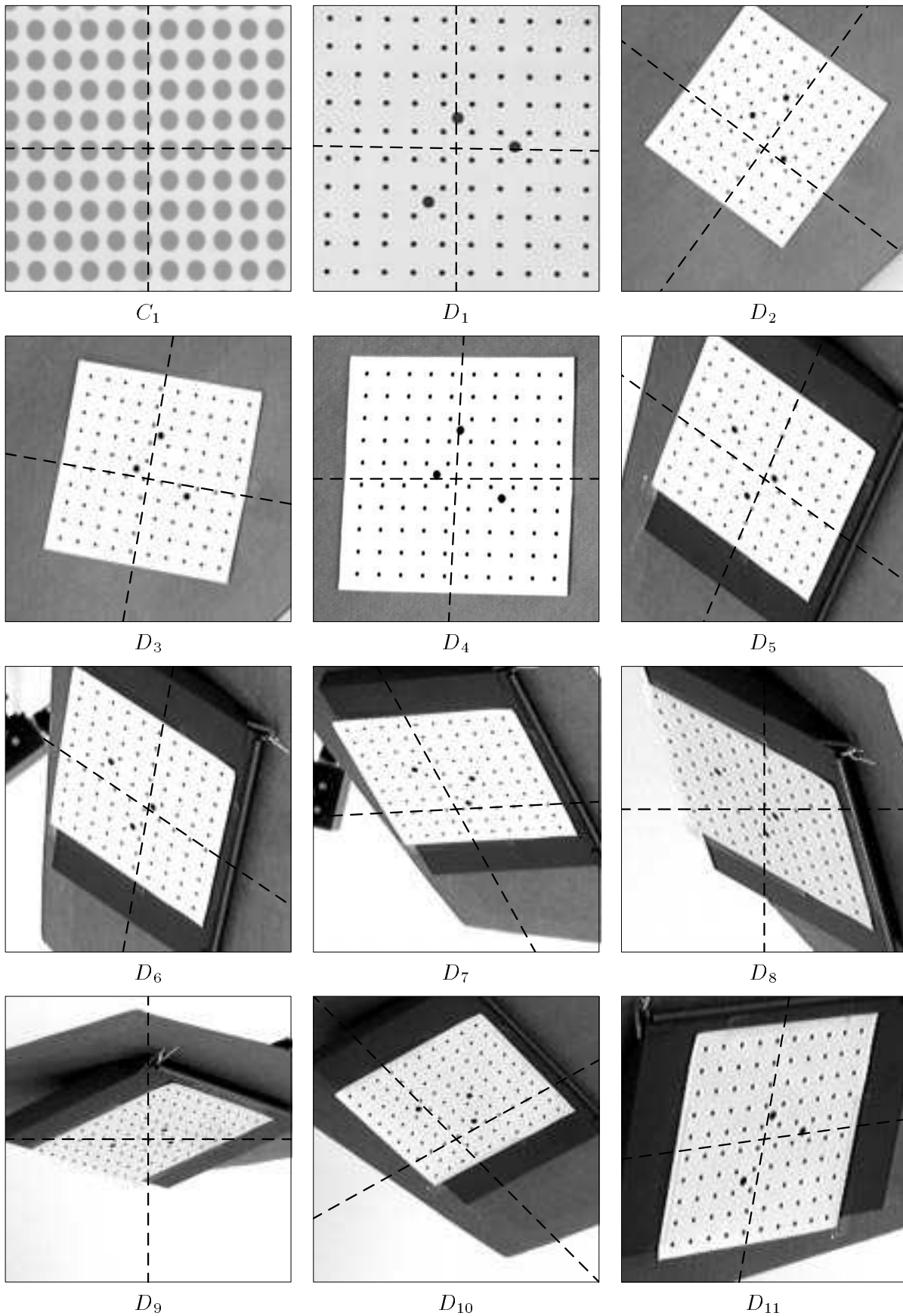


Fig. 48 - Angles obtenus selon la méthode de corrélation angulaire.

5.3 Conclusion

Ce chapitre nous a permis de voir sur un exemple comment le problème de la segmentation se simplifie lorsque les objets à segmenter sont connus.

Cette connaissance permet d'obtenir des algorithmes robustes et précis tout en restant assez efficaces.

L'utilisation conjointe de ces méthodes et de modèles précis des caméras permet d'atteindre des précisions de l'ordre du centième de pixels sur les cas traités dans la littérature (disques, arêtes, coins).

Comme perspectives on peut envisager le suivi en temps réel d'images contenant des grilles, le critère χ pouvant n'être recalculé qu'au voisinage des candidats de l'image précédente et aux positions déduites par la position de la grille.

Conclusion

Contenu de ce document et contributions personnelles

Ce document expose d'une part la conception d'une application chirurgicale et son expérimentation et d'autre part l'étude de problèmes mathématiques et algorithmiques fondamentaux utilisés dans les applications des GMCAO.

Nous présentons tout d'abord la conception et la réalisation d'un système d'assistance à une opération chirurgicale : la rétro-péritonéoscopie.

La conception de ce système comprend la définition d'un protocole de calibrage des instruments, le recalage du patient et la réalisation du geste. Nous avons également conçu des instruments (marqueur "boule" et socle de calibrage) permettant le repérage de l'instrument chirurgical et son calibrage.

La réalisation logicielle du système utilise un ensemble de processus spécialisés partageant les images et les mesures de position. Cette réalisation modulaire permet aisément l'adjonction de fonctionnalités au système. Des travaux sont en cours au laboratoire TIMC afin d'utiliser le système dans d'autres contextes chirurgicaux.

Nous présentons des expériences de validation du système sur des spécimens anatomiques.

Nous proposons ensuite une solution originale au problème de la modélisation de la géométrie et de l'évolution d'un système de solides dans un contexte chirurgical.

Cette étude est basée sur un vaste travail de bibliographie dans les domaines de l'animation et du modelage par ordinateur de la mécanique des milieux continus et des méthodes d'éléments finis.

Nous justifions l'hypothèse d'une évolution quasi-statique du système dans le cadre d'un geste chirurgical. Cette hypothèse permet de ne pas prendre en compte l'inertie des solides et d'éviter ainsi les problèmes de "dérive énergétique" propres aux modèles physiques classiques gérant les contacts et les collisions.

Nous séparons les trois composantes des modèles de systèmes de solides : représentation de la géométrie et des déformations, contraintes et algorithme de contrôle. Ceci nous permet de concevoir un modèle modulaire gérant la cohabitation de solides dont la représentation et les propriétés sont différentes : instruments rigides et organes "mous".

Nous avons utilisé des solides pavés par des éléments paramétriques et détaillons la façon d'associer à chaque solide une énergie potentielle élastique dépendant de sa déformation.

Enfin nous proposons une solution pour traiter le problème des contacts étendus

permanents entre des solides Γ indépendamment de leur représentation et de leur comportement.

Des figures illustrent le fonctionnement du modèle en dimension 2.

Les autres problèmes étudiés ont trait à l'obtention de la précision nécessaire lors de la mise en œuvre du système d'assistance. Nous abordons à travers des cas particuliers les problèmes de la mise en correspondance et de segmentation.

Les problèmes de mise en correspondance et de calibrage nécessitent la manipulation de rotations. Nous consacrons ainsi un chapitre à l'étude de leurs propriétés et représentations.

Nous présentons une synthèse des définitions et propriétés des rotations Γ ainsi que de leurs représentations en dimensions 2 et 3. En particulier Γ dans le cas du passage d'une représentation à une autre en dimension 3 Γ nous donnons des développements permettant d'améliorer la précision des algorithmes existants au voisinage des singularités. Une section porte sur l'étude géométrique des rotations en dimension 4.

Nous étudions la mise en correspondance de nuages de points appariés. Nous exposons et comparons les algorithmes proposés dans la littérature sur ce sujet Γ puis nous proposons une nouvelle approche utilisant la représentation de la rotation inconnue par son "vecteur rotation".

La segmentation est abordée dans le cas de la détection précise de projections d'objets circulaires Γ et de grilles d'objets circulaires. Nous étudions trois facettes de ce problème. Nous voyons d'abord comment localiser grossièrement l'ensemble des objets présents sur une image. Ensuite Γ nous montrons comment on peut raffiner cette information à des résolutions très inférieures au pixel. Enfin Γ nous présentons quelques expériences sur la localisation globale d'une grille d'objets circulaires.

Perspectives

La construction d'un système simulant le comportement des solides concernés par le geste chirurgical est un prolongement naturel de ces travaux. Elle consiste à inclure le modèle de système de solides dans l'application d'assistance.

D'autres applications sont envisageables Γ en particulier la superposition d'images vidéo et d'images de synthèse du modèle informatique Γ et l'interfaçage du système avec des périphériques de "retour d'effort".

L'acquisition des surfaces et des paramètres du comportement des organes peut se faire à l'aide des techniques évoquées dans la conclusion de la deuxième partie : atlas anatomiques Γ mesures directes sur les instruments Γ ajustement d'un modèle sous le contrôle d'un chirurgien. L'utilisation de la vidéo et des systèmes d'acquisition optiques de surfaces permettrait également de suivre les déplacements des organes.

L'étude des composants mathématiques et algorithmiques de base des GMCAO offre des perspectives intéressantes Γ que ce soit pour la conception d'algorithmes efficaces de mise en correspondance Γ l'utilisation des propriétés géométriques des rotations en dimension 4 Γ ou le calibrage automatique de systèmes de vision.

Bibliographie

- [1] *First international symposium on medical robotics and computer-assisted surgery (MRCAS'94)*ΓPittsburghΓPAΓSeptember 1994.
- [2] K.S. ArunΓT.S. HuangΓand S.D. Blostein. Least-squares fitting of two 3-D point sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*Γ9(5):698–700ΓSeptember 1987.
- [3] N. AyacheΓeditor. *Computer vision, virtual reality and robotics in medicine (CVRMED'95)*Γvolume 905 of *Lecture notes in computer science*ΓNiceΓFranceΓApril 1995. Springer.
- [4] E. Bainville. Reconstruction d'objets tridimensionnels à partir de silhouettes. Master's thesisΓEcole Normale Supérieure de LyonΓLyonΓFranceΓJuillet 1992.
- [5] E. BainvilleΓP. ChaffanjonΓand P. Cinquin. Computer generated assistance to a surgical operation : the retroperitoneoscopy. In *MRCAS'94* [1].
- [6] E. BainvilleΓP. ChaffanjonΓand P. Cinquin. Computer generated visual assistance during retroperitoneoscopy. *Computers in biology and medicine*Γ25(2):165–171ΓMay 1995.
- [7] D.H. Ballard. Generalizing the Hough transform to detect arbitrary shapes. *Pattern Recognition*Γ13(2):111–122Γ1981.
- [8] D. Baraff and A. Witkin. Dynamic simulation of non-penetrating flexible bodies. *Computer Graphics (proceedings SIGGRAPH)*Γ26(2)ΓJuly 1992.
- [9] David Baraff. Analytical methods for dynamic simulation of non-penetrating rigid bodies. *Computer Graphics (proceedings SIGGRAPH)*Γ23:223–232ΓJuly 1989.
- [10] David Baraff. Curved surfaces and coherence for non-penetrating rigid body simulation. *Computer Graphics (proceedings SIGGRAPH)*Γ24:19–28ΓAugust 1990.
- [11] David Baraff. Coping with friction for non-penetrating rigid body simulation. *Computer Graphics (proceedings SIGGRAPH)*Γ25(4):31–40ΓJuly 1991.
- [12] David Baraff. Issues in computing contact forces for non-penetrating rigid bodies. *Algorithmica*Γ10:292–352Γ1993.

- [13] David Baraff. Fast contact force computation for non-penetrating rigid bodies. *Computer Graphics (proceedings SIGGRAPH)* pages 23–34 July 1994.
- [14] David Baraff. Interactive simulation of solid rigid bodies. *IEEE Computer Graphics and Applications* 15(3):63–75 May 1995.
- [15] K.J. Bathe. *Finite element procedures in engineering analysis*. Prentice hall 1982.
- [16] D. Bellet and J.J. Barrau. *Cours d'élasticité*. Cepadues-editions 1990.
- [17] Marcel Berger. *Géométrie : espaces euclidiens, triangles, cercles et sphères* volume 2. Cedic/Fernand Nathan 1977.
- [18] Marcel Berger. *Géométrie : la sphère pour elle-même, géométrie hyperbolique, l'espace des sphères* volume 5. Cedic/Fernand Nathan 1977.
- [19] F. Betting J. Feldmar N. Ayache and F. Devernay. A new framework for fusing stereo images with volumetric medical images. In Ayache [3].
- [20] Ph. Bolon J.-M. Chassery J.-P. Cocquerez D. Demigny C. Graffigne A. Montanvert S. Philipp R. Zéboudj and J. Zérubia. *Analyse d'images : filtrage et segmentation*. Enseignement de la physique traitement du signal. Masson 1995.
- [21] Paul Borrel and Ari Rappoport. Simple constrained deformations for geometric modeling and interactive design. *ACM Transactions on Graphics* 13(2):137–155 April 1994.
- [22] C.B. Bose and I. Amir. Design of fiducials for accurate registration using machine vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 12(12):1196–1200 December 1990.
- [23] Armin Bruderlin and Thomas W. Calvert. Goal-directed dynamic animation of human walking. *Computer Graphics (proceedings SIGGRAPH)* 23(3):233–242 July 1989.
- [24] G. Celniker and D. Gossard. Deformable curve and surface finite-elements for free-form shape design. *Computer Graphics (proceedings SIGGRAPH)* 25(4):257–266 July 1991.
- [25] John E. Chadwick David R. Haumann and Richard E. Parent. Layered construction for deformable animated characters. *Computer Graphics (proceedings SIGGRAPH)* 23:243–252 July 1989.
- [26] Ph. Chaffanjon E. Bainville and R. Sarrazin. Place des gestes médicaux-chirurgicaux dans le bilan d'extension ganglionnaire des cancers pelviens. *Bulletin du cancer, Ed. Elsevier* 1995.
- [27] Philippe Chaffanjon. Rétropéritonéoscopie assistée par ordinateur. Dea spécialité génie biologique et médical Université J. Fourier Grenoble France 1993.

- [28] D.T. Chen and D. Zeltzer. Pump it up : computer animation for a biomechanically based model of muscle using the finite element method. *Computer Graphics (proceedings SIGGRAPH)* 26(2) 1992.
- [29] P. G. Ciarlet. *Introduction à l'analyse numérique matricielle et à l'optimisation*. Masson 1982.
- [30] P. G. Ciarlet. *Elasticité tridimensionnelle*. Masson 1985.
- [31] Sabine Coquillart. Extended free-form deformations : a sculpturing tool for 3-D geometric modeling. *Computer Graphics (proceedings SIGGRAPH)* 24:187–196 August 1990.
- [32] S. Cover, N. Ezquerra, J. O'Brien, R. Rowe, T. Gadacz and E. Palm. Interactively deformable models for surgery simulation. *IEEE Computer Graphics and Applications* 13(6) 1993.
- [33] Hervé Delingette, Gérard Subsol, Stéphane Cotin and Jérôme Pignon. A cranofacial surgery simulation testbed. Technical Report 2199 INRIA Sophia-Antipolis France February 1994.
- [34] F. Delmas. Cours de mathématiques supérieures. Lycée Descartes Tours 1988.
- [35] Mathieu Desbrun and M.-P. Gascuel. Animating soft substances with implicit surfaces. *Computer Graphics (proceedings SIGGRAPH)* pages 287–290 August 1995.
- [36] V. Dessenne, S. Lavalée, R. Orti, R. Julliard, S. Martelli and P. Cinquin. Computer assisted knee anterior cruciate ligament reconstruction : first clinical tests. *J. of Image Guided Surgery* 1(1):59–64 1995.
- [37] D. Devernay. Cours de physique de mathématiques spéciales. Lycée Descartes Tours 1989.
- [38] Jean Dieudonné. *La géométrie des groupes classiques*. Springer Verlag 2e edition 1963.
- [39] Jean Dieudonné. *Algèbre linéaire et géométrie élémentaire*. Hermann 1964.
- [40] Fadi Dornaïka. *Contribution à l'intégration vision/robotique : calibrage, localisation et asservissement*. PhD thesis Institut national polytechnique de Grenoble Septembre 1995.
- [41] R.O. Duda and P.E. Hart. Use of the Hough transformation to detect lines and curves in pictures. *Communications of the ACM* 15(1):11–15 January 1972.
- [42] G. Dumont, B. Arnaldi and G. Hégron. Mechanics of solids for computer animation. In *Proceedings PIXIM 89* 1989.

- [43] G. Duvaut. *Mécanique des milieux continus*. MassonΓ1990.
- [44] A. Efrat and C. Gotsman. Subpixel image registration using circular fiducials. *International Journal of Computational Geometry and Applications*Γ4(4):403–422ΓDecember 1994.
- [45] T. ErtlΓH. RuderΓR. AllrutzΓK. GruberΓM. GuntherΓF. HospachΓM. RuderΓJ. SubkeΓand K. Widmayer. Interactive control of biomechanical animation. *The Visual Computer*Γ9(8)Γ1993.
- [46] O.D. Faugeras. Quelques pas vers la vision artificielle en trois dimensions. *Technique et Science Informatiques*Γ7(6):547–590Γ1988.
- [47] O.D. Faugeras and M. Hebert. The representationΓrecognition and locating of 3-D objects. *International Journal of Robotics Research*Γ5(3):27–52Γ1986.
- [48] R. Featherstone. The calculation of robot using articulated-body inertia. *International journal of robotics research*Γ2(1):13–30Γ1983.
- [49] J.-D. Gascuel and M.-P. Gascuel. Displacement constraints for interactive modeling and animation of articulated structures. *The Visual Computer*Γ10(4):191–204Γ1994.
- [50] M.-P. Gascuel. *Déformations de surfaces complexes: techniques de haut niveau pour la modélisation et l'animation*. PhD thesisΓUniversité Paris-SudΓcentre d'OrsayΓOctobre 1990.
- [51] M.-P. Gascuel. An implicit formulation for precise contact modeling between flexible solids. *Computer Graphics (proceedings SIGGRAPH)*ΓAugust 1993.
- [52] M.-P. GascuelΓA. VerroustΓand C. Puech. A modeling system for complex deformable bodies suited to animation and collision processing. *Journal of Visualization and COmputer Animation*Γ2:82–91Γ1991.
- [53] H. Gié and J.P. Sarmant. *Mécanique, tome 1*. Tech. et Doc. LavoisierΓ1985.
- [54] H. Gié and J.P. Sarmant. *Mécanique, tome 2*. Tech. et Doc. LavoisierΓ1985.
- [55] Andrew S. GlassnerΓeditor. *Graphics gems*. Academic PressΓ1990.
- [56] J.-P. Gourret. Modeling 3-D contacts and deformations using finite elements theory in synthetic human tactile perception. In *SIGGRAPH 88 course notes on synthetic actors*Γ1988.
- [57] J.-P. GourretΓN. Magnenat ThalmannΓand D. Thalmann. Simulation of object and human skin deformations in a grasping task. *Computer Graphics (proceedings SIGGRAPH)*Γ23(3)ΓJuly 1989.
- [58] W.E.L. GrimsonΓG.J. EttingerΓS.J. WhiteΓP.L. GleasonΓT. Lozano-PerezΓW.M. Wells IIIΓand R. Kikinis. Evaluating and validating an automated registration system for enhanced reality visualization in surgery. In Ayache [3].

- [59] J.K. Hahn. Realistic animation of rigid bodies. *Computer Graphics (proceedings SIGGRAPH)* 22(4) August 1988.
- [60] R.M. Haralick, Hyonam Joo, Chung nan Lee, Xinhua Zhuang, V.G. Vaidya and Man Bae Kim. Pose estimation for corresponding point data. In H. Freeman (editor) *Machine Vision for Inspection and Measurement* 1989.
- [61] C.J. Henri, A.C.F. Colchester, J. Zhao, D.J. Hawkes, D.L.G. Hill and R.L. Evans. Registration of 3-D surface data for intra-operative guidance and visualization in frameless stereotactic neurosurgery. In Ayache [3].
- [62] C.-T. Ho and L.-H. Chen. A fast ellipse/circle detector using geometric symmetry. *Pattern Recognition* 28(1):117–124 January 1995.
- [63] Radu Horaud and Olivier Monga. *Vision par ordinateur. Outils fondamentaux*. Hermes 1993.
- [64] B.K.P. Horn. Closed-form solution of absolute orientation using unit quaternions. *J. Opt. Soc. Amer. A* 4(4):629–642 April 1987.
- [65] B.K.P. Horn, H.M. Hilden and S. Negahdaripour. Closed-form solution of absolute orientation using orthonormal matrices. *J. Opt. Soc. Amer. A* 5(7):1127–1135 July 1988.
- [66] Donald H. House and David E. Breen. Particles as modeling primitives for surgical simulation. In *Proceedings of the 11th international conference of the IEEE Engineering in Medicine and Biology Society* volume 3 1989.
- [67] J. Illingworth and J. Kittler. A survey of the Hough transform. *Computer Vision, Graphics and Image Processing* 44:87–116 1988.
- [68] A. Jokhadar, C. Bard and C. Laugier. Planning dextrous operations using physical models. In *IEEE International conference on robotics and automation* May 1994.
- [69] François Jouve. *Modélisation de l'œil en élasticité non linéaire*. PhD thesis Université Paris VI 1992.
- [70] Kenichi Kanatani. Analysis of 3-D rotation fitting. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 16(5):543–549 May 1994.
- [71] C. Kimme, D. Ballard and J. Sklansky. Finding circles by an array of accumulators. *Communications of the ACM* 18(2):120–122 February 1975.
- [72] S. Kumar, N. Ranganathan and D. Goldgof. Parallel algorithms for circle detection in images. *Pattern Recognition* 27(8):1019–1028 1994.
- [73] J.O. Lachaud and E. Bainville. A discrete adaptative model following topological modifications of volumes. In *Discrete Geometry for Computer Imagery* Grenoble France September 1994.

- [74] S. Lavalée and R. Szeliski. Recovering the position and orientation of free-form objects from image contours using 3-D distance maps. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 17(4):378–390 1995.
- [75] Zexiang Li and John Canny. Motion of two rigid bodies with rolling constraint. *IEEE Transaction on Robotics and Automation* 6(1):62–72 February 1990.
- [76] Zhiping Li and Martin Reed. A finite element method to model progressive fracturing. *Computer Methods in Applied Mechanics and Engineering* 120(3–4):303–313 1995.
- [77] M.C. Lin and D. Manocha. Fast interference detection between geometric models. *The Visual Computer* 11:542–561 1995.
- [78] Z.C. Lin T.S. Huang S.D. Blostein H. Lee and E.A. Margerum. Motion estimation from 3-D point sets with and without correspondences. In *Proc. Conference on Computer Vision and Pattern Recognition, Miami Beach, FL* pages 194–201 June 1986.
- [79] Jean-Christophe Lombardo. *Modélisation d’objets déformables avec un système de particules orientés*. PhD thesis Université J. Fourier Grenoble II January 1996.
- [80] Annie Luciani. *Un outil informatique de création d’images animées: modèles d’objets, langage, contrôle gestuel en temps réel. Le système ANIMA*. PhD thesis Institut National Polytechnique de Grenoble 1985.
- [81] P.J. Mercier. Cours de physique de mathématiques supérieures. Lycée Descartes Tours 1988.
- [82] Gavin S.P. Miller. The motion dynamics of snakes and worms. *Computer Graphics (proceedings SIGGRAPH)* 22(4) August 1988.
- [83] Michel Minoux. *Programmation mathématique. Volume 1*. Dunod 1989.
- [84] Michel Minoux. *Programmation mathématique. Volume 2*. Dunod 1989.
- [85] Rached Mneimné and Frédéric Testard. *Introduction à la théorie des groupes de Lie classiques*. Hermann 1986.
- [86] J. Moisan. Cours de mathématiques spéciales. Lycée Descartes Tours 1989.
- [87] M. Moore and J. Wilhelms. Collision detection and response for computer animation. *Computer Graphics (proceedings SIGGRAPH)* 22(4) August 1988.
- [88] A. Norton G. Turk B. Bacon J. Gerth and P. Sweeney. Animation of fracture by physical modeling. *The Visual Computer* 7(4):210–218 1991.
- [89] R. Orti S. Lavalée R. Julliard P. Cinquin and E. Carpentier. Computer assisted knee ligament reconstruction. In *IEEE Engineering in Medicine and Biology Society Proceedings* pages 936–937 San Diego 1993.

- [90] A. Pentland and J. Williams. Good vibrations : modal dynamics for graphics and animation. *Computer Graphics (proceedings SIGGRAPH)* 23(3):215–222 July 1989.
- [91] B. Peuchot, A. Tanguy and M. Eude. Virtual reality as an operative tool during scoliosis surgery. In Ayache [3].
- [92] John C. Platt and Alan H. Barr. Constraint methods for flexible models. *Computer Graphics (proceedings SIGGRAPH)* 22(4):279–288 August 1988.
- [93] Ari Rappoport, Yaacov Hel-Or and Michael Werman. Interactive design of smooth objects with probabilistic point constraints. *ACM Transactions on Graphics* 13(2):156–176 April 1994.
- [94] W.T. Reeves. Particle systems – a technique for modeling a class of fuzzy objects. *ACM Transactions on Graphics* 2(2):91–108 April 1983.
- [95] A. Reverchon and M. Ducamp. *Mathématiques sur micro-ordinateur : analyse* volume 1. Eyrolles 1984.
- [96] R. Sarrazin, J.-F. Dyon, A. Vernay and P. Coquilhat. La rétroperitonéoscopie. *La lettre chirurgicale* May 1990.
- [97] P. Sautot, P. Cinquin, S. Lavallée and J. Troccaz. Computer assisted spine surgery : a first step towards clinical application in orthopaedics. In *IEEE Engineering in Medicine and Biology Society Proceedings* pages 1071–1072 Paris November 1992.
- [98] Peter Schröder and David Zeltzer. The virtual erector set : dynamic simulation with linear recursive constraint propagation. *Computer Graphics, special issue on 1990 Symposium on interactive 3-D graphics* 24(2):23–31 March 1990.
- [99] Sederberg and Parry. Free form deformation of solid geometric models. *Computer Graphics (proceedings SIGGRAPH)* 20(4) 1986.
- [100] Ken Shoemake. Animating rotation with quaternion curves. *Computer Graphics (proceedings SIGGRAPH)* 19(3):245–254 July 1985.
- [101] R. Szeliski and D. Tonnesen. Surface modeling with oriented particle systems. Technical Report CRL 91/14 DEC Cambridge research laboratories 1991.
- [102] R.H. Taylor, S. Lavallée, G.C. Burdea and R. Mösges. *Computer-Integrated Surgery. Technology and clinical applications*. MIT Press Cambridge MA London England 1995.
- [103] D. Terzopoulos and K. Fleischer. Modeling inelastic deformations : viscoelasticity, plasticity, fracture. *Computer Graphics (proceedings SIGGRAPH)* 22(4):269–278 August 1988.
- [104] D. Terzopoulos, J. Platt, A. Barr and K. Fleisher. Elastically deformable models. *Computer Graphics (proceedings SIGGRAPH)* 21(4):205–214 1987.

- [105] Demetri Terzopoulos and Hong Qin. Dynamic NURBS with geometric constraints for interactive sculpting. *ACM Transactions on Graphics* 13(2):103–136 April 1994.
- [106] R. Théodor. *Initiation à l'analyse numérique*. Masson 1989.
- [107] J. Troccaz, Y. Menguy, M. Bolla, P. Cinquin, P. Vassal, N. Laieb and S.D. Soglio. Patient set-up optimization for external conformal radiotherapy. In *MRCAS'94* [1].
- [108] M. Uenohara and T. Kanade. Vision-based object registration for real-time image overlay. In Ayache [3].
- [109] Shinji Umeyama. Least-squares estimation of transformation parameters between two point patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 13(4):376–380 April 1991.
- [110] C.W.A.M. van Overveld. A technique for motion specification in computer animation. *The Visual Computer* 6:106–116 1990.
- [111] Jane Wilhelms. Towards automatic motion control. *IEEE Computer Graphics and Applications* 7(4):11–22 April 1987.
- [112] Andrew Witkin and Michael Kass. Spacetime constraints. *Computer Graphics (proceedings SIGGRAPH)* 22(4) August 1988.
- [113] Andrew Witkin and William Welch. Fast animation and control of non-rigid structures. *Computer Graphics (proceedings SIGGRAPH)* 24:243–252 August 1990.
- [114] R.K.K. Yip, P.K.S. Tam and D.N.K. Leung. Modification of Hough transform for circles and ellipses detection using a 2-dimensional array. *Pattern Recognition* 25(9):1007–1022 September 1992.
- [115] J. Zhao, A.C.F. Colchester, C.J. Henri, D.J. Hawkes and C. Ruff. Visualization of multimodal images for neurosurgical planning and guidance. In Ayache [3].
- [116] O.C. Zienkiewicz and R.L. Taylor. *The finite element method: basic formulation and linear problems* volume 1. McGraw Hill 4th edition 1989.
- [117] O.C. Zienkiewicz and R.L. Taylor. *The finite element method: solid and fluid mechanics dynamics and non-linearity* volume 2. McGraw Hill 4th edition 1991.
- [118] O.C. Zienkiewicz and R.L. Taylor. *La méthode des éléments finis: formulation de base et problèmes linéaires*. AFNOR technique 1991.