



HAL
open science

Modélisation et Simulation de la Respiration

Emmanuel Promayon

► **To cite this version:**

Emmanuel Promayon. Modélisation et Simulation de la Respiration. Modélisation et simulation. Université Joseph-Fourier - Grenoble I, 1997. Français. NNT : . tel-00130070

HAL Id: tel-00130070

<https://theses.hal.science/tel-00130070v1>

Submitted on 9 Feb 2007

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.

THÈSE

Présentée par

Emmanuel PROMAYON

Pour obtenir le titre de docteur de
l'Université Joseph Fourier – Grenoble 1
(arrêtés ministériels du 5 juillet 1984 et du 30 mars 1992)

(Spécialité INFORMATIQUE)

Modélisation et Simulation de la Respiration

Date de la soutenance :	21 novembre 1997	
Composition du jury :	Jacques DEMONGEOT	Président
	Dominique BECHMANN	Rapporteur
	Daniel THALMANN	Rapporteur
	Pierre BACONNIER	Directeur
	Claude PUECH	Co-directeur

Thèse préparée au sein du laboratoire TIMC-IMAG.

La mécanique du tronc est un problème simple dans son essence et devient compliqué seulement de par la nature variée et complexe des structures et matériaux composant le tronc. L'application sensée des principes d'ingénierie et des techniques informatiques nous permettront de maîtriser ces détails et de comprendre comment le tronc fonctionne.

S.H. Loring, E.N. Bruce, (Loring et Bruce, 1986)

Remerciements

Je tiens à remercier...

PIERRE BACONNIER, sa disponibilité et ses conseils ont été d'une grande aide dans mon travail de thèse. Son approche m'a toujours été très enrichissante.

CLAUDE PUECH, pour avoir accepté d'être le co-directeur de cette thèse.

PHILIPPE CINQUIN qui accepta en 1992 de m'accueillir au sein de l'équipe Gestes Médico-Chirurgicaux Assistés par Ordinateur.

DOMINIQUE BECHMANN et DANIEL THALMANN, pour l'intérêt qu'ils ont porté à mon travail et les remarques qui ont contribuées à l'amélioration de ce document.

JACQUES DEMONGEOT, pour m'avoir accueilli au laboratoire Traitement d'Images Modélisation et Cognition, d'avoir accepté de faire partie du jury et pour l'attention qu'il a portée à ce travail.

Je remercie aussi les membres de l'équipe GMCAO, permanents aussi bien qu'étudiants, ainsi que MARIBEL CHENIN pour leurs aides et leurs soutiens tout au long de cette thèse.

...

Remerciements “particuliers”

Il me reste encore à remercier tous ceux qui m’ont élevé et supporté pendant toutes ces années d’études. La liste est longue, pourtant chacune de ces personnes mérite toute ma gratitude. Parce qu’une thèse ce n’est pas que 3 ans de travail :

Yvonne et Guy, Florence (“mon seul double au monde”); Loreline (♥), Océane (la mirgnette), Florian (l’eskeëllent Florian), Pascale et Jean-Christophe (dans l’ordre inverse d’apparition dans mon/le monde); Gillou (M^e Gilles), Riquet (MC Marcel), Steflet (Guinness Brother), Jean-Marc (gouverneur Mamac), (dans l’ordre usküllométrique, Comic Metal is the law); Tof, Stephane (M^e Chiwawa), Laurent, Jean-Charles, (avec Riquet, les destructeurs de Taërik); BB (colo-dictateur), Juju (géniale), Sergio (trop fort), GG (bzz-bzz) et Vindss (ping-pong man) (PPRistiquement votre); Arleen (la rousse), Ruthie (la brune), Sharon (la blonde), (les Guinness Sisters !); Jek et Adrien (avec Gillou, “mes amis” des heures heureuses); et enfin Delphine (mon *ex*-secrétaire particulière)...

Merci aussi à tous ceux qui sont venus me supporter de leurs amitiés en ce lendemain mémorable du beaujolais nouveau 1997 (accessoirement, jour de ma soutenance). Je pense notamment à Bernt & Silke, Bobone, Colette et Michel, Cyril, Djibi, Edith, Isabella, Jean-Baptiste & Quentin & Marie-Laure, Kristof, Laure, Maurice & Colette, Nadège & ManuU, Nico & Véro & Bruno, Pierral, PJ, Rabaa (la 4^{ème}) et Santi, Scro (l’ambassadeur) & Sévrine, Titouan & Léo & Benoit & Sandrine, Vincent, Walter, X-tine,... Que la farce soit avec vous (et avec votre esprit).

Vous entrez maintenant dans le monde merveilleux du...



Table des matières

INTRODUCTION	1-5
1. Modélisation et simulation	1
2. Cadre et motivation.....	3
3. Contributions et organisation du document.....	4
Partie 1 : étude du phénomène et représentation des connaissances	4
Partie 2 : le modèle informatique	5
Partie 3 : étude et validation qualitative du modèle informatique	5
PARTIE 1	7-58
CHAPITRE 1 : LE SYSTÈME VENTILATOIRE	7-30
1. Aperçu du système mécanique ventilatoire	8
1.1 Historique.....	8
1.1.1 De l'antiquité jusqu'au XVème siècle	8
1.1.2 Les muscles respiratoires et leurs actions.....	8
1.1.3 Vue intégrale de la mécanique ventilatoire	8
1.2 Les composants.....	9
1.3 Description sommaire d'un cycle respiratoire	10
1.4 Description anatomique des composants	10
1.4.1 Le diaphragme.....	10
1.4.2 La cage thoracique et ses muscles	11
1.4.3 L'abdomen.....	12
1.4.4 Les poumons.....	12
1.4.5 Autres muscles	12
1.5 Conclusion	13
2. Mesurer les propriétés du système ventilatoire	13
2.1 Principes fondamentaux	13
2.2 Les instruments de mesure	14
2.2.1 Mesures des quantités physiologiques	14
2.2.2 Mesures cinétiques	14
3. Comprendre	15
3.1 Comportement statique	15
3.1.1 Études	15
3.1.2 Résultats	16
3.2 Comportement mécanique passif de la cage thoracique	16
3.2.1 Études	16
3.2.2 Résultats	17
3.3 Dynamique de la respiration.....	18

3.3.1 Équations du mouvement.....	18
3.3.2 Quantifier l'action dynamique des différents composants.....	19
4. Les muscles moteurs de la respiration.....	20
4.1 Concepts fondamentaux.....	20
4.1.1 Les fibres	20
4.1.2 Propriétés mécaniques.....	21
4.1.3 Propriétés électriques.....	22
4.2 Propriétés individuelles.....	22
4.2.1 Considérations générales.....	22
4.2.2 Diaphragme.....	23
4.2.3 Muscles intercostaux.....	23
4.2.4 Sternomastoidiens et scalènes	23
4.2.5 Abdominaux	24
4.2.6 Autres muscles.....	24
4.3 Action des muscles respiratoires	24
4.3.1 Diaphragme.....	24
4.3.2 Cage thoracique	26
4.3.3 Abdomen.....	27
4.3.4 Modèle mécanique de l'action musculaire	27
5. Conclusion	28

CHAPITRE 2 : SIMULATION PHYSIQUE EN INFOGRAPHIE	31-58
--	--------------

1. Principes généraux de la mécanique des corps solides et méthodes d'intégration	33
1.1 Dynamique du point	33
1.2 Dynamique des objets rigides	34
1.3 Représentation de l'orientation des solides.....	36
1.4 Méthode d'intégration.....	37
1.4.1 Schéma d'Euler :.....	38
1.4.2 Schéma de Newton-Cotes.....	38
1.4.3 Schéma de Runge-Kutta.....	38
2. Les modèles déformables.....	39
2.1 Approches nodales	40
2.1.1 Discrétisation d'objets	40
2.1.2 Assemblage d'éléments.....	43
2.1.3 Conclusion	46
2.2 Approches globales	48
2.2.1 Déformations globales	48
2.2.2 Déformations modales	48
3. Contraindre les modèles.....	49
3.1 Contraindre.....	49
3.2 Les différentes méthodes de résolution.....	50
3.2.1 Méthodes d'optimisation.....	51
3.2.2 Méthodes de recherche des forces exercées par les contraintes.....	51
3.2.3 Méthodes de recherche des déplacements dus aux contraintes	52
4. La problématique actuelle	53

4.1 Simulation d'objets complexes hétérogènes	53
4.1.1 Approche transversale	54
4.1.2 Plate-forme	54
4.2 Base géométrique des objets simulés	55
4.3 Paramétrisation des modèles	55
4.4 Autres problèmes	55
4.4.1 Modélisation des muscles	56
4.4.2 Stabilité (méthode d'intégration vs modèle informatique)	56
4.4.3 Interactivité (problématique GMCAO)	56
4.4.4 Simulation de l'humain dans des mondes virtuels	57
4.4.5 Gestion des contacts, collisions, et changements topologiques	57
5. Conclusion.....	57

PARTIE 2	59-120
-----------------	---------------

CHAPITRE 3 : CONSTRUCTION DU MODÈLE	59-88
--	--------------

1. Organisation du chapitre	60
2. Description générale	61
2.1 Géométrie des objets déformables	61
2.1.1 Modélisation surfacique des objets	61
2.1.2 Utilisation de facettes triangulaires	61
2.2 La dynamique du système	62
2.2.1 Les forces	63
2.2.2 Principe d'action-réaction	63
2.2.3 Algorithme d'évolution	64
2.2.4 Récapitulatif sur les propriétés générales des éléments :	65
3. Régions élastiques	66
3.1 Formulation de la forme et de la déformation	66
3.1.1 Théorie de l'élasticité	66
3.1.2 Tenseurs de déformation surfacique	68
3.1.3 Repère local basé sur le voisinage	69
3.2 Mémoire de forme locale	72
3.3 Analyse de l'apport de la mémoire de forme	73
3.3.1 Comparaison formelle	73
3.3.2 Comparaison empirique : simulation	79
3.3.3 Conclusion	81
3.4 Récapitulatif sur les éléments et régions élastiques	82
4. Régions musculaires	82
4.1 État de l'art	82
4.1.1 Simulation des déformations	82
4.1.2 Simulation du fonctionnement	83
4.1.3 Conclusion	84
4.2 Modèle de force musculaire	84
4.3 Récapitulatif sur les éléments et régions musculaires	86

5. Régions solides	86
5.1.1 Récapitulatif sur les éléments et régions solides.....	86
6. Conclusion	86

CHAPITRE 4 : CONTRAINDRE LE MODÈLE	89-120
---	---------------

1. Raccordement des régions	89
1.1 Raccordement entre deux régions élastiques.....	90
1.2 Raccordement entre une région élastique et une région solide.....	90
1.2.1 Méthode.....	91
1.2.2 Algorithme	91
1.3 Raccordement entre deux régions solides	92
1.4 Conclusion.....	93
2. Résolution des contraintes.....	93
2.1 Formulation dynamique de la résolution de contrainte	94
2.2 Principe de la résolution directe	95
2.2.1 Algorithme	95
2.2.2 Choix de la direction.....	96
2.2.3 Résolution	98
2.2.4 Avantages	99
2.2.5 Inconvénients	99
2.2.6 Contrôle supplémentaire et détection des zones de discontinuité.....	100
2.3 Comparaison	101
2.3.1 Problème	101
2.3.2 Multiplicateur de Lagrange (Platt et Barr, 1988).....	102
2.3.3 Optimisation dans le sous-espace orthogonal (Fua et Bechbühler, 1997).....	104
2.3.4 Méthode de résolution directe.....	106
2.3.5 Conclusion	108
2.4 Multi-contraintes.....	109
3. Contraintes locales et contraintes globales.....	109
3.1 Contraintes locales.....	109
3.2 Contraintes globales	110
4. Une contrainte globale : l'incompressibilité	110
4.1 État de l'art	111
4.2 Fonction de volume	112
4.3 Le gradient de la contrainte.....	115
4.4 Le vecteur de projection	116
4.5 Généralisation	118
4.5.1 Validité physique	118
4.5.2 Compressibilité	119
4.5.3 Autres modèles	119
5. Conclusion	120

1. Implémentation	122
1.1 Développement et interface	122
1.2 Implémentation du modèle	122
1.2.1 Architecture.....	122
1.2.2 Données.....	122
1.2.3 Manipulation	123
2. Exemples	124
2.1 Deux ballons connectés	124
2.1.1 Modèle.....	124
2.1.2 Simulation.....	125
2.1.3 Conclusion.....	127
2.2 Bras	127
2.2.1 Modèle.....	127
2.2.2 Simulation.....	128
2.2.3 Conclusion.....	130
2.3 Conception d'objets à volume constant	130
2.3.1 Déformation à volume constant	131
2.3.2 Déformation interactive à volume constant.....	131
2.3.3 Conclusion.....	133
3. Modèle du tronc	133
3.1 Bibliographie.....	133
3.2 Modèle simplifié.....	134
3.2.1 Modèle.....	134
3.2.2 Simulation.....	135
3.2.3 Conclusion.....	136
3.3 Construction du modèle	136
3.4 Étude séparée des structures composant le modèle complet.....	137
3.4.1 La cage thoracique.....	138
3.4.2 L'ensemble composé par l'abdomen et diaphragme	139
3.5 Simulation	141
3.5.1 Conclusion.....	142
4. Comparaison et validation	143
4.1 Paramètres à mettre en correspondance.....	144
4.2 Données cinématiques : mouvements du tronc	147
4.2.1 Mesure des mouvements par système optique 3D	147
4.2.2 Interfacer notre modèle.....	148
4.3 Conclusion	148

1. Contribution.....	151
2. Perspectives	152

ANNEXES **155-174**

**ANNEXE 1 : IMPLÉMENTATION DE LA DYNAMIQUE D'UN OBJET SOLIDE
(SYSTÈME DE POINTS MATÉRIELS)** **155-162**

- 1. Données de l'objet 156
- 2. Initialisations 156
 - 2.1 Calcul du centre des masses 156
 - 2.2 Calcul de la matrice d'inertie et de la matrice et du quaternion d'orientation 157
 - 2.3 Calcul des coordonnées des points dans le repère 158
 - 2.4 Initialisation des autres variables 158
- 3. Simulation d'une itération 158
 - 3.1 Calcul des nouvelles grandeurs (à l'instant) 159
 - 3.2 Intégration, calcul de la nouvelle position..... 160
 - 3.2.1 Schéma d'Euler..... 160
 - 3.2.2 Schéma de Newton-Cotes..... 160
 - 3.2.3 Nouvelle position et nouvelle vitesse..... 160

ANNEXE 2 : UTILISATION DES QUATERNIONS **163-166**

- 1. Passage de la matrice de rotation au quaternion..... 163
- 2. Passage du quaternion à la matrice de rotation..... 165
- 3. Multiplication de deux quaternions 165

ANNEXE 3 : NORMALES **167-170**

ANNEXE 4 : FORMAT DU MODÈLE **171-174**

- 1. Description du format : 171
- 2. Exemple..... 173

BIBLIOGRAPHIE **175-185**

Introduction

Comprendre et analyser l'inconnu sont des comportements ancrés au plus profond de la conscience humaine. Ces comportements sont exacerbés lorsque c'est le propre fonctionnement de l'humain qui est en jeu. Un des processus de représentation cognitive des phénomènes naturels, passe par l'intermédiaire de la création de schémas mentaux. Cette représentation mentale est souvent simplifiée mais qualifie, dans ses grandes lignes, le phénomène observé. La science permet d'affiner cette représentation et de déterminer ses caractéristiques aussi bien quantitatives que qualitatives. On aboutit alors à un modèle dont l'objectif est de simuler le phénomène représenté. Cette introduction va nous permettre d'élaborer le processus directeur de ce manuscrit.

1. Modélisation et simulation

Dans le cadre de sa représentation cognitive, un phénomène biologique est vu au travers de nos sens comme agissant par l'intermédiaire et sur des corps, « systèmes vivants ». Cette perception est traduite dans un modèle cognitif dans lequel les corps mis en jeu sont des objets que l'on manipule. Ce modèle est de plus caractérisé par les objectifs qu'il veut atteindre. Pour cela, il ne retient qu'un certain nombre de lois de comportement des objets manipulés et ne prend en compte qu'un certain nombre de scénarios (c'est-à-dire un certain nombre de situations à analyser ou expliquer). Il en va de même pour le modèle mathématique ou pour le modèle informatique (voir Figure 1). On peut donc considérer le modèle comme un outil de représentation, reflet du monde réel. Cet outil, en utilisant des objets représentatifs des corps, va simuler leurs comportements au travers de scénarios.

La Figure 1 représente un exemple de représentation d'un phénomène au travers d'un modèle cognitif. Ce modèle cognitif se construit autour d'un outil d'analyse. Ici on ne considère que les modélisations informatiques. Le modèle à proprement parlé est un ensemble de moteurs agissant à partir de lois de comportement et de représentations de situations sur des objets représentatifs des corps modélisés. Lorsqu'un phénomène dynamique est modélisé de la sorte, le modèle informatique fait appel à une boucle de simulation.

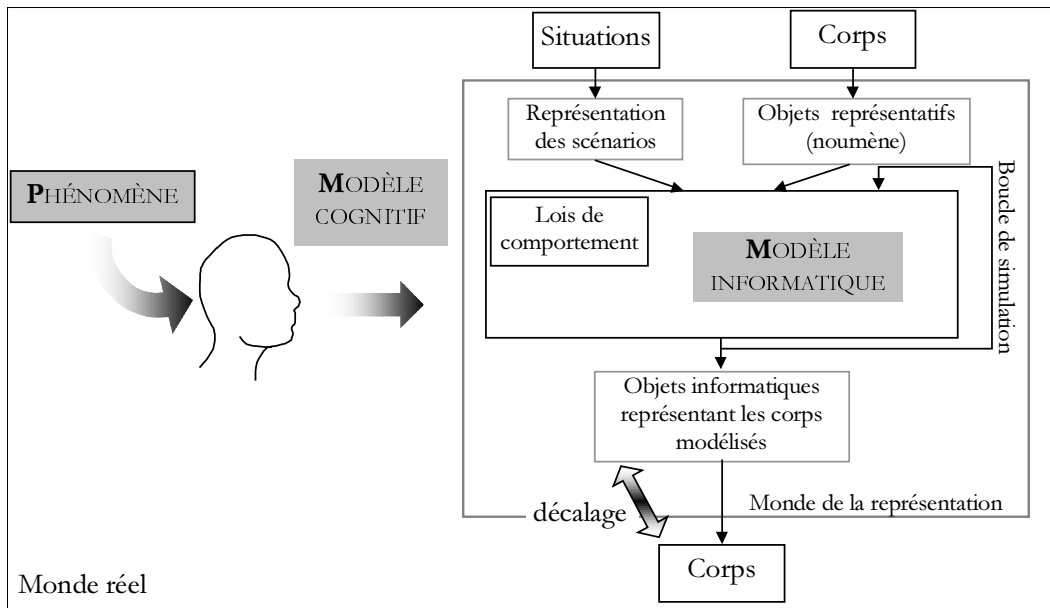


Figure 1. Modélisation informatique.

La modélisation des systèmes biologiques et l'utilisation de mesures réelles pour estimer des valeurs de paramètres sont des outils potentiellement puissants pour l'analyse et la compréhension de ces systèmes. Pourtant, la phase de modélisation doit être effectuée avec soin. En effet, une fois le modèle construit, il sera difficile de dégager ce qui est une erreur due à la modélisation (simplification trop grande du phénomène) de ce qui provient plutôt d'une erreur interne du modèle (mauvaise représentation). Le modèle permet de réduire le nombre de données brutes pour affiner l'analyse d'un phénomène, mais lorsque le modèle est trop simple son utilité est amoindrie.

Un modèle ayant un nombre réduit de paramètres ne peut prendre en compte toutes les données (situations et corps) possibles. Il est important de connaître les limites de réalisme d'un modèle informatique. Notre démarche se doit donc aussi de déterminer la limite au-delà de laquelle il y a décalage entre la perception du phénomène et le résultat du modèle (voir Figure 1).

De ces remarques découle une méthodologie plus générale pour la modélisation, c'est celle que j'ai suivie tout au long de cette thèse :

- **Étape 1 :** choisir les structures internes et les principes gouvernant le comportement du phénomène à modéliser.
- **Étape 2 :** construire le modèle à partir des mécanismes à modéliser.
- **Étape 3 :** étudier le comportement du modèle en fonction d'un petit nombre de paramètres et de données laissés libres (les autres étant fixes) pour obtenir une première validation qualitative de la modélisation.

Lorsque l'adéquation entre modélisation et objectifs est atteinte, le modèle peut être affiné en repartant de l'étape 1 pour le complexifier. Pour cela, les éléments ignorés sont examinés en profondeur dans le modèle simple, le niveau d'analyse est alors redéfini.

Il se peut aussi que les principes fondamentaux soient bien analysés, mais que, malgré tout, la discrétisation et la distribution de ces principes en un nombre fini d'unités introduise une erreur. Dans ce cas le besoin de contrôle sur le modèle doit être plus important. Le modèle doit être contraint plus fortement.

Selon l'objectif défini lors de la première étape, les sorties (conséquences du modèle, objets représentatifs) peuvent varier. Dans le cadre de cette thèse, nous nous limiterons aux conséquences visibles (objets graphiques représentatifs). Notre modélisation se fixe pour but d'obtenir une sortie graphique animée aussi proche que possible de la réalité (géométriquement et cinétiquement). L'objectif de cette thèse est donc la fabrication d'un outil, sous forme de modèle informatique, permettant d'atteindre ce but, et la validation qualitative de cet outil.

2. Cadre et motivation

Le phénomène que nous cherchons à modéliser, de manière plus particulière, est la respiration de l'être humain. De nombreuses modalités aujourd'hui disponibles permettent une analyse tridimensionnelle des structures mécaniques de la respiration : l'imagerie par résonance magnétique et la tomодensitométrie permettent l'analyse géométrique des structures du thorax, l'acquisition vidéo de marqueurs placés à la surface de la peau permet la surveillance et le monitoring des mouvements du tronc. L'interprétation de telles données nécessite l'utilisation de nouveaux modèles. Un tel modèle, une fois établi et validé, permettrait de fournir des diagnostics non invasifs de certaines pathologies.

Comme nous nous attachons à la représentation graphique du phénomène, notre but devient la modélisation des mouvements du tronc pendant la respiration. Cette modélisation passe par le développement d'un système interactif de manipulation des objets (simulation).

La modélisation des mouvements du tronc sera présentée comme la modélisation et la simulation des mouvements d'un objet complexe (ensemble formé par le diaphragme, la cage thoracique, les poumons et l'abdomen).

Cet objet complexe, composé de parties et régions aux propriétés et comportements différents, est issu :

- de données anatomiques fixes (ne faisant intervenir que des positions relatives de structures en mouvement),
- de données fonctionnelles (propriétés mécaniques passives et actives, normales ou pathologiques),
- de situations ou scénarios faisant intervenir des contraintes appliquées au système (position du sujet, ventilation assistée, parole, marche...).

La motivation d'une telle étude est la production de séquences animées du tronc au cours de la respiration afin d'en comprendre le phénomène.

Un autre objectif se situe dans le cadre plus général de la simulation de l'humain à l'intérieur de mondes virtuels et/ou interactifs (Badler, 1997), (Thalmann, 1996). La représentation virtuelle de personnages humains a pour objectif la simulation de la présence de ces humains à des fins d'ergonomie, de conception, ou de communication. Aujourd'hui le réalisme de ces

personnages virtuels est devenu primordial. Il y a quelques années encore une simulation était jugée bonne lorsqu'elle était « graphiquement correcte ». On veut maintenant aller plus loin : il s'agit de construire des personnages dont les comportements sont individuels et autonomes, qui sont capables de réagir à leur environnement. Dans ce cadre, on s'oriente actuellement vers des simulations comportementales. Le modèle informatique développé au cours de cette thèse se veut une contribution à cette problématique.

On peut encore définir d'autres objectifs, à plus long terme :

- couplé à un appareil d'acquisition d'images médicales par l'intermédiaire de capteurs, le modèle permettrait de piloter l'acquisition à partir de la phase respiratoire,
- couplé à des capteurs de position, le modèle pourrait servir au recalage (mise en correspondance) de différentes modalités d'imagerie (dans le cadre de la problématique des gestes médico-chirurgicaux assistés par ordinateur (Cinquin et al, 1995)),
- en surveillance dans des unités de soin intensif, le modèle permettrait de minimiser l'erreur commise par des capteurs de type tomographie à impédance électrique en fournissant une mesure de l'expansion de la cage thoracique au cours des mesures. (Adler et al, 1996) considère que l'erreur des capteurs due à cette expansion varie actuellement entre 2 et 20%.

3. Contributions et organisation du document

Ce manuscrit suit dans ses grandes lignes l'ordre chronologique dans lequel cette thèse s'est préparée. Cet ordre correspond aussi aux étapes successives du processus de modélisation décrit au paragraphe 1, les trois étapes se traduisant par les trois parties de ce manuscrit.

Partie 1 : étude du phénomène et représentation des connaissances

La première étape suit un cheminement logique répondant successivement aux questions suivantes :

- Comment fonctionne le système réel ?
- Quels sont les principes, mécanismes et conséquences du phénomène à insérer dans le schéma de modélisation (définition des objectifs) ?
- De quelles données réelles dispose-t-on ? Quels degrés de pertinence peut-on leur attribuer ?
- Quels sont les paramètres à prendre en compte pour atteindre les objectifs ?
- Quels sont les objets informatiques représentatifs à utiliser ou fabriquer pour atteindre les objectifs ?

Ce cheminement est proposé comme méthodologie générale pour la modélisation informatique de phénomènes. L'étude du phénomène physiologique, afin d'en dégager les notions et les mécanismes essentiels, est présentée au chapitre 1. On y verra qu'un modèle comportemental tridimensionnel prenant en compte un ensemble de contraintes (telle que l'incompressibilité) est nécessaire. Cela nous amènera à fixer le cadre des recherches des outils informatiques à utiliser : les modèles générateurs basés sur la physique. Le chapitre 2 livre alors une étude des concepts mécaniques simplifiés et des outils informatiques de déformation proposés dans la littérature infographique des modèles générateurs.

Partie 2 : le modèle informatique

Le processus de modélisation se poursuit dans la deuxième partie par la construction du modèle proprement dit. Le chapitre 3 présente les bases du modèle informatique développé au cours de cette thèse. Notre approche est de considérer un objet comme étant constitué de régions ayant des comportements différents et nécessaires à la construction de comportements tridimensionnels complexes. L'aspect général de ces objets est tout d'abord présenté. Ces objets étant constitués de régions élastiques, de régions musculaires et de régions solides (indéformables), ces trois aspects sont successivement exposés. La définition d'un principe généraliste considérant une force engendrée par un point d'attraction est essentielle dans ce chapitre. La forme des objets est représentée de manière originale par l'intermédiaire de trois paramètres locaux. Ces paramètres permettent de proposer une nouvelle fonction pour la modélisation de l'élasticité utilisant une mémoire de forme locale. Comparée à une méthode classique (masse-ressort), elle prouve son efficacité (tant au niveau de la stabilité qu'au niveau de la modélisation). Dans un deuxième temps, les régions musculaires sont définies. Ces régions reproduisent, de manière simple, une base comportementale de simulation des muscles (contrôle de l'intensité de la force motrice, de l'activation, et des directions de contractions). Les régions solides sont présentées en dernière partie du chapitre 3.

La description du modèle se poursuit par l'étude, au chapitre 4, des contraintes que l'on doit utiliser pour que le modèle puisse reproduire tous les comportements distingués dans la partie 1. Pour nous, il s'agit de modéliser des caractéristiques physiques, comme l'incompressibilité, par l'intermédiaire de contraintes sur les objets.

Dans ce chapitre, on développe une nouvelle méthode de résolution directe de contraintes permettant de vérifier des contraintes locales et globales sans utiliser de méthode itérative. La résolution de la contrainte d'incompressibilité illustre cette méthode de résolution. La généralisation de notre méthode de résolution de l'incompressibilité permet de dégager des principes utilisables dans d'autres domaines.

Partie 3 : étude et validation qualitative du modèle informatique

Au cours de la dernière partie (chapitre 5), nous fournirons une validation qualitative du comportement du modèle grâce à de nombreux exemples d'objets. Nous y développons un modèle du tronc et de ses mouvements au cours de la respiration.

La conclusion générale fournit un récapitulatif des points originaux de ce document ainsi qu'une ouverture sur les perspectives de la modélisation informatique.

Chapitre 1

Le système ventilatoire

Il faut comprendre pour mesurer et non mesurer pour comprendre.

Gaston Bachelard

Ce chapitre va nous permettre de comprendre le phénomène de la respiration. Nous allons examiner comment ce phénomène est généré. Tout au long de cet examen nous nous attacherons à dégager des éléments et hypothèses pour la construction du modèle. Ce chapitre entre donc directement dans la phase de perception et d'analyse de la modélisation.

Dans un premier temps, nous explorons globalement la physiologie respiratoire. Un historique des développements de cette science permet d'établir les premiers jalons dans la compréhension du système. Une analyse des composantes principales du système respiratoire nous fournit ensuite un découpage élémentaire. Nous passons en revue les principaux instruments de mesure disponibles : ce sont moyens de valider nos hypothèses sur le modèle et les paramètres du modèle. Nous voyons ensuite plus en détail les caractéristiques du système ventilatoire (aussi bien statiques que dynamiques). Nous nous attachons enfin à décrire les muscles moteurs de la respiration.

En conclusion, nous établissons les bases du modèle.

Les bases bibliographiques de ce chapitre peuvent être trouvées dans (Macklem et Mead, 1986), (Roussos et Macklem, 1985), (Taylor et al., 1989), et (West 1995).

1. Aperçu du système mécanique ventilatoire

La physiologie de la ventilation doit répondre à la question suivante : comment l'air entre-t-il et sort-il des poumons ? Elle s'inspire à la fois de l'arrangement et de l'action individuelle des muscles (d'après l'observation et l'anatomie) et de lois physiques (grâce à des mesures de forces, de pressions, de volumes et de leurs changements au cours du temps).

1.1 Historique

Nous allons brièvement tracer un historique des découvertes dans le domaine de la mécanique respiratoire. On ne reprendra ici que les avancées majeures de cette science, pour plus de détails, se référer à (Otis, 1986). À travers cet historique nous découvrirons les éléments essentiels à prendre en compte dans la modélisation.

1.1.1 De l'antiquité jusqu'au XV^{ème} siècle

L'étude du système ventilatoire n'est pas une science récente, Erasistratus de Chios, considéré comme le père de la physiologie, écrit en 300 avant JC que « *le diaphragme est le seul muscle responsable de la respiration* ». Galen (~130-200 après JC) puis J. Mayow (1640-1679) font la remarque que « *le poumon suit le mouvement du thorax* ». R. Descartes (1596-1650) influence fortement la direction des recherches en énonçant le principe que « *la matière vivante est sujette aux lois physiques* » (l'homme est une machine).

1.1.2 Les muscles respiratoires et leurs actions

F. Magendie (1783-1855) démontre que « *l'abdomen agit comme un levier en réponse à l'action du diaphragme* ». Sir A. Keith (1866-1955) marque le début de la description qualitative et précise de l'action des muscles respiratoires.

1.1.3 Vue intégrale de la mécanique ventilatoire

F. Rohrer (1868-1926) est le premier à fournir une formulation quantitative et abstraite de la mécanique respiratoire. W. O. Fenn (1893-1971) marque le début de l'ère moderne en élaborant les relations entre pression et volume du système respiratoire, et la physiologie des muscles.

L'importance des découvertes doit être mise en parallèle avec les mécanismes essentiels à la modélisation (les principes fondamentaux sont ceux dont les conséquences sont les plus observables). Grâce aux progrès des techniques d'observation (appareils de mesure, et imagerie médicale), la physiologie respiratoire a pu continuellement approfondir sa connaissance. Notre modèle devra donc prendre en compte les découvertes fondamentales sur le plan mécanique et systématique décrites ci-dessus. Nous allons maintenant décomposer le tronc en éléments actifs afin d'en comprendre le fonctionnement.

1.2 Les composants

Avant de rentrer dans les détails, considérons le système ventilatoire composé par quatre éléments principaux (Taylor et al., 1989) :

- une pompe :
 - le diaphragme,
 - la cage thoracique et ses muscles,
 - l'abdomen (et ses muscles).
- Un échangeur de gaz : les poumons.

Les différents volumes pulmonaires sont présentés à la Figure 1.

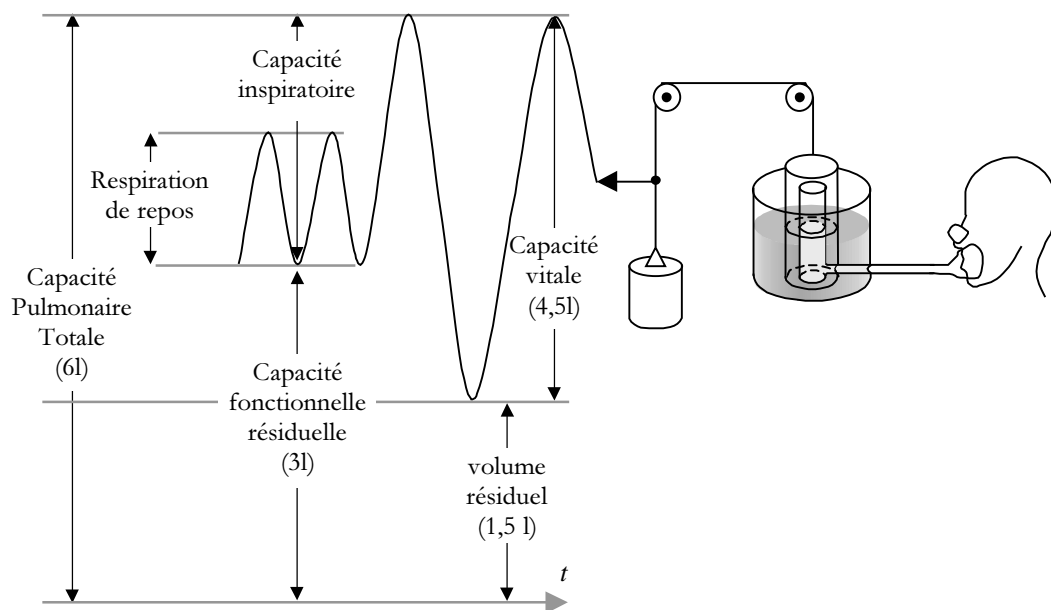


Figure 1. Dynamique des échanges gazeux du poumon au cours des différentes phases de la respiration. Les quantités indiquées en litres sont des mesures moyennes. Les mesures se font en respirant au travers d'une cloche, en manœuvres d'inspirations et d'expirations forcées.

Voyons d'abord comment agissent ces différents composants au cours d'un cycle respiratoire.

1.3 Description sommaire d'un cycle respiratoire

La mécanique respiratoire étudie les forces qui déforment les poumons et déplacent la paroi thoracique ainsi que les résistances qui s'opposent à ces forces. Deux phases sont distinguées :

- l'inspiration : le muscle principal de l'inspiration est le diaphragme. Au cours d'une respiration normale le niveau du diaphragme subit des déplacements de l'ordre du centimètre (West 1995). En cas d'exercice et de manœuvres forcées, d'autres muscles se mettent en action et notamment les muscles scalènes et sterno-cléido-mastoïdiens (plus communément appelés stérnomastoïdiens) élevant les deux premières côtes et le sternum (l'amplitude des déplacements diaphragmatiques peut dans ce cas-là atteindre 10 cm).
- l'expiration : l'expiration de repos est un événement passif. L'énergie élastique stockée dans le système pendant l'inspiration est généralement suffisante pour produire une expiration normale sans besoin supplémentaire d'énergie (comme un ballon se dégonflant). Le poumon et la cage thoracique tendent à revenir à leur position d'équilibre après avoir subi une expansion active au cours de l'inspiration. Cependant, là aussi, en cas d'exercice ou d'une demande importante du système (comme dans certains types de maladies pulmonaires), d'autres muscles viennent accélérer ce mouvement (muscles de la ceinture abdominale, muscles intercostaux). L'expiration devient alors active.

Chacun de ces composants a des propriétés géométriques et physiques (statiques ou dynamiques) différentes. Examinons la géométrie du système avant de passer à l'observation et aux mesures de sa dynamique (au paragraphe 2).

1.4 Description anatomique des composants

1.4.1 Le diaphragme

Le diaphragme est un muscle plat, fin et musculotendineux, (Perrotin et Moreaux, 1965).

Ses fibres s'élèvent de deux sources principales :

- partie crurale (vertébrale) : de la troisième vertèbre lombaire et arches aponeurotiques (sans insertion dans la partie postérieure des côtes) ;
- partie costale : de la surface interne supérieure des six côtes les plus basses.

Elles sont dirigées vers le haut, parallèlement à l'axe du corps et sont opposées directement à la surface interne de la partie inférieure de la cage thoracique. Elles convergent pour former le tendon central (seuls les mammifères et quelques espèces d'oiseaux ont une séparation complète entre les cavités thoraciques et abdominales).

La forme du diaphragme est un cylindre coiffé d'un dôme (Figure 2).

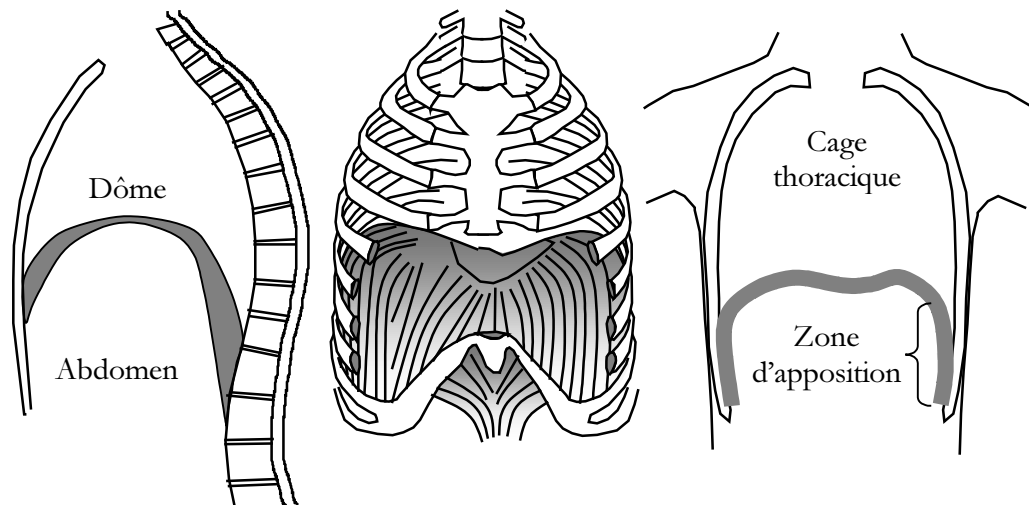


Figure 2. Position et forme du diaphragme (en coupe de profil, en perspective de devant et en coupe de face).

1.4.2 La cage thoracique et ses muscles

La cage thoracique est constituée, pour ce qui nous préoccupe, d'une cage osseuse (formée par les côtes : 10 paires de côtes reliées au sternum par des cartilages et deux paires de côtes dites « flottantes », voir Figure 3) et de muscles.

Parmi les muscles, nous ne considérons que les muscles intercostaux, les muscles scalènes et les muscles sternomastoïdiens.

Les muscles intercostaux forment deux plans fins de fibres musculaires occupant tout l'espace intercostal. On distingue les intercostaux internes et externes par leur disposition par rapport à la surface du corps. Tous deux ont des fibres obliques et orientées vers le bas, mais alors que les externes sont orientés vers l'avant de la côte supérieure à la côte inférieure, les internes sont orientés vers l'arrière.

Outre les intercostaux, un certain nombre de muscles participent sous certaines conditions à l'acte respiratoire. Parmi tous ces muscles, appelés « muscles accessoires » deux seulement montrent une activité significative : les scalènes et les sternomastoïdiens. Les scalènes élèvent la cage thoracique pendant l'inspiration. Ils sont souvent utilisés lors de la respiration de repos et ne doivent probablement pas être considérés comme des muscles accessoires. Ils partent des cinq vertèbres cervicales les plus basses et s'étendent jusqu'à la surface supérieure de la première et seconde côte.

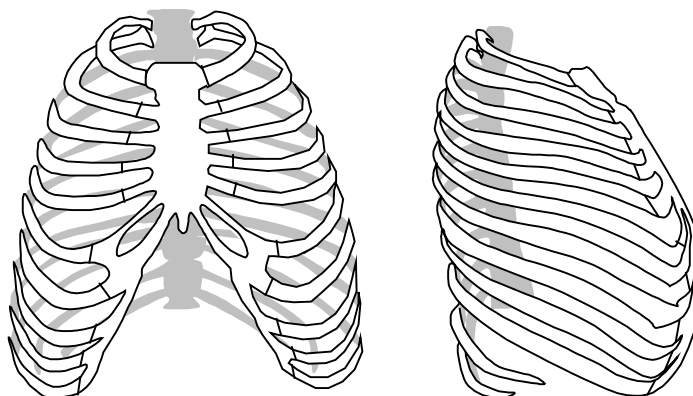


Figure 3. La cage thoracique : partie osseuse de face et de profil.

1.4.3 L'abdomen

Les muscles de l'abdomen que nous considérons sont essentiellement ceux de la paroi antéro-latérale (obliques interne et externe, transversus abdomini et rectus abdomini).

Des études récentes montrent qu'en plus de faciliter l'expiration, les abdominaux joueraient un rôle dans l'inspiration. En se contractant, ils compressent le contenu abdominal et maintiennent les fibres du diaphragme dans des étirements plus grand (ce qui permet d'optimiser la courbe de longueur/tension du muscle, voir Figure 9, paragraphe 4.1). Le même phénomène intervient lors de changement de posture couché/debout et lors de certains exercices.

1.4.4 Les poumons

Les poumons jouent un rôle pour l'échange gazeux, mais aussi pour la parole et le chant ; ils ont des effets cardio-vasculaires, et des fonctions métaboliques.

Les poumons sont enfermés dans les cavités pleurales. Chaque plèvre se décompose en deux couches de membranes remplies de liquide. Chaque poumon possède une élasticité considérable. Ils s'adaptent à la paroi de la cavité thoracique. Leurs mouvements dépendent donc de l'action musculaire au niveau du tronc.

Ils se gonflent et se dégonflent environ 15 fois/minute, avec une moyenne de 500ml/respiration (ainsi chaque jour les poumons ventilent au moins 10.800 litres d'air, 5.700 litres de sang ; 600 litres d'oxygène et 460 litres de dioxyde de carbone sont ainsi transférés).

1.4.5 Autres muscles

Il existe d'autres muscles du tronc connus ou considérés par certains comme ayant des actions respiratoires (les trapèzes, les pectoraux majeurs et mineurs, les serrati antérieurs, postérieurs supérieurs et postérieurs inférieurs, les sacrospinalis, les levatores costarum, les transversus thoracis subcalius, les vitissimus dorsi et les quadratus lumborum).

Ils ne seront pas traités dans ce document.

1.5 Conclusion

Le besoin se fait sentir de modèles structurels qui imitent les principales caractéristiques des relations entre les pressions, les forces passives et les changements de géométrie de l'ensemble de la cage thoracique.

(Smith et Loring, 1986)

À la lecture de la littérature, on se rend compte qu'historiquement une évolution technique a toujours précédé une évolution des découvertes. Aujourd'hui, les techniques d'imagerie et de mesures sont de plus en plus fines. Pourtant, les physiologistes n'ont pas encore de moyens d'appréhender et d'assimiler toutes les données disponibles. Il manque actuellement un outil capable de fournir une vue d'ensemble mais aussi raffinée de la mécanique ventilatoire.

Avant d'examiner les différentes propriétés des éléments structurels de la cage thoracique nous allons voir maintenant comment nous pouvons mesurer ces propriétés.

2. Mesurer les propriétés du système ventilatoire

Nous décrivons ici les principes et les instruments de mesure disponibles afin de montrer les moyens de valider un modèle mais aussi d'expliquer les besoins des physiologistes.

2.1 Principes fondamentaux

Les physiologistes se servent des mesures pour déterminer les origines, les conséquences et les significations d'un phénomène. C'est sur la phase de mesures que repose toute l'analyse. Pourtant, la mesure est elle-même, par principe, entachée d'erreur :

- la connexion d'un appareil de mesure : elle peut entraîner une modification de la mesure,
- la réponse de l'instrument : l'appareil de mesure a des paramètres intrinsèques qu'il faut estimer,
- le traitement du signal : les techniques de traitement du signal ont pour objet de faciliter l'interprétation des mesures mais elles alourdissent parfois le processus de mesure,
- l'interprétation des mesures : elle est fortement liée à la présence et à la justesse d'un modèle.

En ce qui nous concerne, les mesures vont nous servir dans deux cas. De manière directe elles fournissent une estimation des paramètres et de manière indirecte un moyen de valider des hypothèses d'analyse.

2.2 Les instruments de mesure

Nous devons connaître a priori les moyens de construire et de vérifier nos hypothèses de modélisation. Comme la démarche classique du physiologiste, celle du modélisateur informatique doit prendre en compte les mesures disponibles. La liste succincte des différentes techniques utilisées pour mesurer les grandeurs physiques du système respiratoire est séparée ici en deux types : celles mesurant les quantités physiologiques (variation de pression, de volume ou d'activité musculaire) et celles liées aux propriétés géométriques ou cinétiques (dimensions, déplacements).

2.2.1 Mesures des quantités physiologiques

Les mesures physiologiques disponibles concernent les variations de volume et de pression globale ou locale du tronc (grâce à des spiromètres, des capteurs de pression, des jauges de contrainte), les indications de commande (mesurées par ENG : électroneurographies), et d'activités musculaires (en utilisant des EMG, électromyographie et appareils de mesure de courbe force/tension).

2.2.2 Mesures cinétiques

Deux types de mesures cinétiques doivent être distingués : les mesures internes et les mesures externes.

Les mesures internes sont les plus difficiles à obtenir. Elles concernent les déplacements des structures internes. On les obtient par IRM et par tomographie à rayon X. La phase d'analyse et de segmentation des mouvements internes est délicate et peu de données sont disponibles dans la littérature.

Les systèmes de mesure externe sont essentiellement des systèmes optiques analysant les déplacements de marqueurs positionnés à la surface du tronc (Ferrigno et al, 1990). Nous verrons dans la partie 3 de ce document les différentes méthodes existantes et celles que nous utiliserons par la suite pour la validation de notre modèle : (Ferrigno et al, 1994) et (Carnevali et al, 1996).

L'intérêt de telles méthodes est qu'en mesurant les dimensions du tronc on peut aussi étudier les variations de volume de la cage thoracique et de l'abdomen nécessaires à l'étude de la dynamique du tronc. Les mouvements et les déformations du tronc au cours des phases respiratoires nous donnent certaines indications sur les actions et les activités des muscles respiratoires spécifiques ou de groupes de muscles, voir notamment (Loring et Bruce, 1986) et (De Troyer et Loring, 1986).

De plus, les mesures externes, donc non invasives, sont moins traumatisantes. Un autre intérêt du modèle 3D serait de permettre une visualisation des mouvements internes à partir de ces mesures.

À partir de cet ensemble de mesures nous allons maintenant analyser le système respiratoire de manière plus précise afin de dégager les éléments importants pour la construction de notre modèle.

3. Comprendre

Une bonne compréhension est nécessaire avant toute modélisation. Une erreur dans l'analyse risque d'induire une erreur dans la modélisation. Nous examinerons le comportement statique, mécanique passif et enfin dynamique du tronc au cours de la respiration.

3.1 Comportement statique

Pendant la relaxation des muscles respiratoires sous conditions statiques, les pressions exercées par le système respiratoire dépendent des différentes forces qui agissent sur les poumons, la cage thoracique, le diaphragme et l'abdomen : les forces élastiques, de surface et d'accélération (comme la gravité).

3.1.1 Études

Le comportement statique est étudié à travers les relations pression/volume. L'expérience la plus commune est d'inspirer à travers un spiromètre un certain volume, puis d'expirer (ou de relaxer les muscles) par une voie obstruée. La mesure des pressions exercées dans cette voie donne une mesure du comportement statique. Les mesures de la Figure 4 sont effectuées en variant petit à petit le volume pulmonaire.

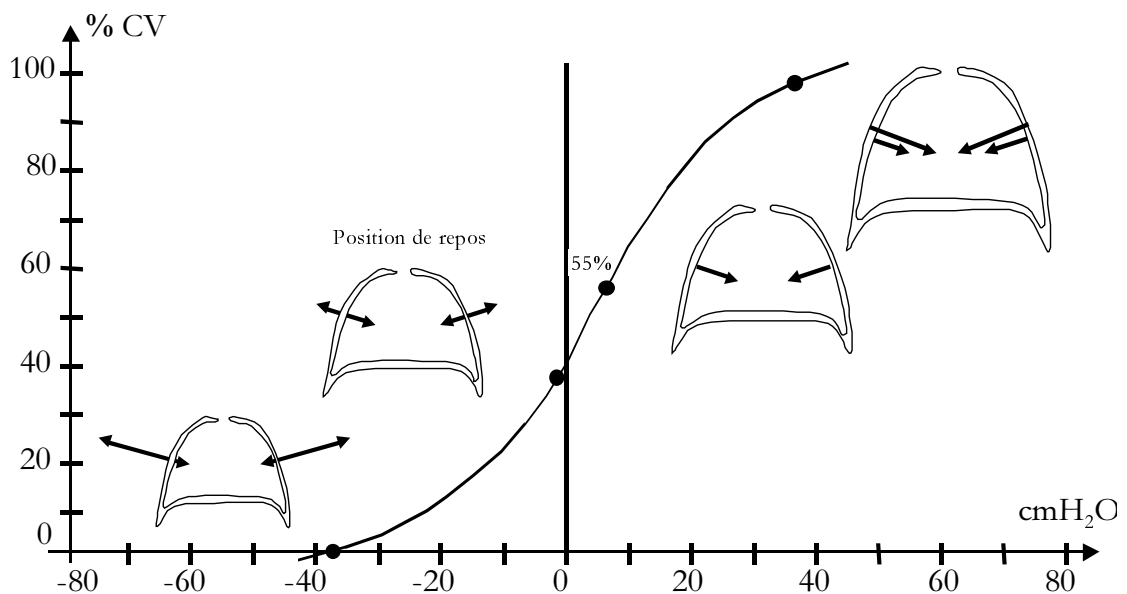


Figure 4. Courbe de pression/volume statique du système respiratoire, adaptée de (Agostini et Hyatt, 1986). Le volume est donné en pourcentage de la capacité vitale (CV). Les flèches représentent les forces statiques relatives exercées sur le système par la cage thoracique et le poumon.

On peut aussi étudier les effets de changement de posture et de position. Ces changements sont importants pour l'abdomen (influencé fortement par la gravité). Chez un sujet debout, les muscles respiratoires relaxés, l'abdomen se comporte comme un récipient dont une paroi serait extensible. La pression dans la partie supérieure est négative. Couché, le diaphragme n'a aucune activité dans la phase d'expiration grâce à la compensation abdominale (due à la gravité).

3.1.2 Résultats

L'étude du comportement statique du système nous permet d'établir quelques remarques :

- Le volume de repos du poumon est inférieur à 0% de la capacité vitale (c'est-à-dire moins que le volume résiduel). Celui de la cage thoracique est de 55% de la capacité vitale.
- De plus, si on ouvre l'espace pleural sur l'air extérieur, les poumons se collapent complètement. *Les poumons sont donc fortement élastiques.*
- L'étude des changements de posture nous procure deux renseignements importants : l'abdomen est mécaniquement équivalent à un récipient rempli de liquide. Comme le volume des gaz de l'abdomen chez l'humain est négligeable par rapport au volume pulmonaire, *le volume de l'abdomen est considéré comme constant lors de changement de pression.*
- Les relations statiques pression/volume ne dépendent pas seulement du volume mais aussi de son historique. Au cours de certains exercices, on voit apparaître ce qu'on appelle des boucles d'hystérèse : le système n'arrive pas à suivre le même trajet à l'aller (quand les forces sont appliquées) et au retour (relaxation). Ce phénomène est commun et on le retrouve dans de nombreux tissus structurels du corps et de matériaux non biologiques (comme les métaux rigides et les bandes élastiques). C'est une constatation importante puisqu'elle met en évidence *l'importance des forces viscoélastiques et plastiques.* L'hystérèse est négligeable dans la respiration de repos (heureusement d'ailleurs puisque l'aire de la courbe d'hystérèse peut être considérée comme la perte d'énergie du système).

Ces constatations sont à prendre en compte dans le modèle.

3.2 Comportement mécanique passif de la cage thoracique

Le comportement mécanique du tronc est déterminé par les propriétés contractiles actives des muscles respiratoires mais aussi par les propriétés passives des structures formant le tronc. C'est l'interaction des éléments passifs et actifs qui détermine les forces et les déformations du tronc. C'est pourquoi pour construire notre modèle nous avons besoin d'étudier le comportement mécanique passif de la cage thoracique, du diaphragme et de l'abdomen.

3.2.1 Études

Le tronc est un arrangement de parois élastiques interconnectées, chacune ayant des propriétés distinctes. Pour caractériser ces propriétés, on utilise l'anatomie et l'observation externe.

Du point de vue anatomique, on distingue la paroi de la cavité thoracique, la paroi abdominale et le diaphragme délimitant ces deux cavités (le diaphragme chez un homme moyen couvre la moitié de la surface interne du tronc en état de volume résiduel). L'observation externe ne distingue que deux éléments seulement : la cage thoracique et l'abdomen.

Quand la posture est fixe et que les forces appliquées sont relativement petites, le tronc a deux degrés de liberté : le déplacement de la cage thoracique et le déplacement de la paroi abdominale (Figure 5).

3.2.2 Résultats

On fait ici plusieurs constats concernant les éléments du tronc et leurs caractéristiques mécaniques :

- La densité moyenne du contenu abdominal est similaire à celle de l'eau. Ainsi sous conditions statiques, les parois sont sujettes à un gradient de pression/surface approximant un *gradient hydrostatique* (Smith et Loring, 1986).

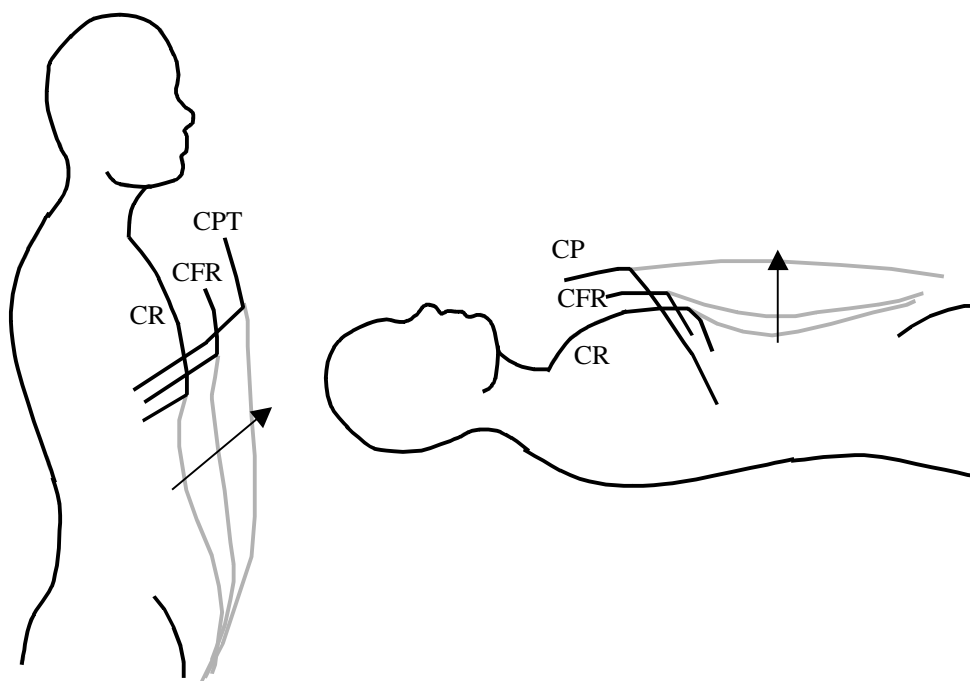


Figure 5. Profils de la paroi abdominale illustrant les configurations d'équilibre à différents niveaux de volume pulmonaire : capacité résiduelle (CR), capacité fonctionnelle résiduelle (CFR) et capacité pulmonaire totale (CPT), pour deux positions debout et allongé sur le dos (d'après (Smith et Loring, 1986)).

- Le diaphragme est une structure composite où les éléments musculaires s'étendent des vertèbres, des côtes et du sternum jusqu'au tendon central aponeurotique. Le tendon central est une feuille relativement fine d'une tension très forte composée par un réseau de fibres cartilagineuses occupant une fraction considérable de l'aire surfacique du dôme diaphragmatique. Bien que très peu de données soient aujourd'hui disponibles, *on considère le tendon comme cent fois moins extensible que la partie musculaire à proprement parler* (Smith et Loring, 1986). Ainsi on peut dire que la contraction du diaphragme ne peut pas se faire de manière homogène.

- Pour un volume donné, les pressions de rappel élastiques pendant l'expansion de la cage thoracique sont supérieures à celles de la contraction. Ces problèmes d'hystérèse nous amènent à constater que *la cage thoracique n'est pas un système parfaitement élastique*.

Les caractéristiques mécaniques du tronc résultent d'interactions complexes entre ses différents éléments structurels. L'équilibre mécanique est du à une distribution largement répartie des pressions et des forces. Chacune des structures majeures du tronc a une géométrie distincte et un ensemble de propriétés physiques doivent être déterminées pour comprendre le comportement de la structure composite.

Il y a encore trop peu d'information quantitative aussi bien sur les propriétés intrinsèques des structures individuelles que sur les forces qui agissent sur les différentes parois, mais aussi sur les forces impliquées dans les interactions structurelles. Le grain des connaissances n'étant pas fixé, le modèle que nous devons construire devra permettre d'ajuster les propriétés mécaniques globales à différents niveaux de précision et de géométrie.

3.3 Dynamique de la respiration

Les forces, contrant les propriétés élastiques inertielles et de résistance au flux des poumons et de la cage thoracique, entraînent un mouvement du système respiratoire produisant ainsi la ventilation des poumons. Si on excepte la ventilation mécanique, ces forces sont produites par les muscles respiratoires. Nous allons étudier maintenant la dynamique du système avant d'examiner plus en détail les muscles et leurs actions (au paragraphe 4).

3.3.1 Équations du mouvement

La Figure 6 décrit l'équation du mouvement de la mécanique classique, pour un mobile posé sur un plan, attaché à un ressort et un amortisseur.

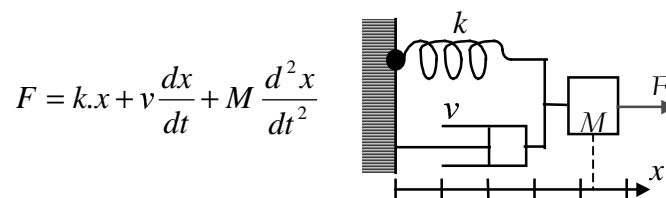


Figure 6. Équation du mouvement de la mécanique classique. F est la somme des forces appliquées au système, x la position du mobile, k la constante de raideur du ressort, v le coefficient de frottement, et M la masse du mobile monté sur ressort.

Par analogie, voir (Rodarte et Rehder, 1986), on peut poser la même équation pour le système respiratoire (Figure 7).

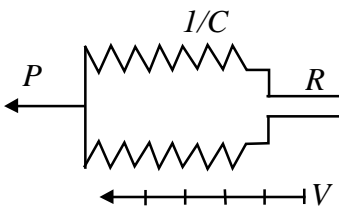
$$P = \frac{1}{C}V + R\frac{dV}{dt} + I\frac{d^2V}{dt^2}$$


Figure 7. Analogie de l'équation du mouvement pour le système respiratoire. P représente la pression générant le flux de gaz dans les poumons, V est le volume du soufflet, C la compliance du système, R la résistance au flux et I l'inertie relative à l'accélération de la pression.

Le système respiratoire est considéré comme un soufflet relié à l'extérieur par un tube de résistance non nulle au flux. On peut aussi faire des analogies électriques, de nombreux problèmes de la mécanique respiratoire pouvant être approchés en utilisant les solutions électriques pour prévoir les réponses mécaniques (c'est pourquoi on présente la compliance de la même façon que la capacité des équations électriques).

Comme le système respiratoire n'est pas un système linéaire, la résistance et la compliance ne sont pas constantes. On a déjà vu qu'elles étaient dépendantes de l'historique du volume et du flux. Le système respiratoire est en effet constitué d'embranchements complexes de voies aériennes représentant des résistances en séries et en parallèles connectées à des éléments élastiques des parenchymes pulmonaires, qui à leur tour sont connectés en série avec le tronc, composé lui-même de différentes structures de la cage thoracique, du diaphragme et de l'abdomen. Malgré tout, ce modèle simple est une base de travail utile pour examiner le comportement dynamique du système ventilatoire normal.

3.3.2 Quantifier l'action dynamique des différents composants

Comme nous l'avons vu au paragraphe 1.4, les muscles principaux de l'inspiration sont le diaphragme, les intercostaux parasternaux, les scalènes, les sternomastoïdiens, les intercostaux externes et d'autres muscles accessoires (utilisés dans des ventilations importantes). Les abducteurs des cordes vocales et les muscles de l'hypopharynx sont aussi des muscles inspiratoires importants car ils jouent un rôle dans le maintien des ouvertures des voies aériennes.

Les relations entre longueur, tension et vitesse de contraction des muscles d'une part et volumes, pressions et flux du système respiratoire d'autre part sont complexes. Ainsi, les pressions maximales pouvant être générées par les muscles diminuent si les raccourcissements des muscles franchissent une valeur optimale. Ou encore, pour une longueur donnée, la force musculaire diminue proportionnellement à l'augmentation de la vitesse de contraction. Ainsi les pressions maximales pouvant être générées par les muscles sont déterminées à la fois par le volume pulmonaire et le flux de gaz.

Nous allons maintenant examiner en détail les actions musculaires générant la dynamique du système ventilatoire.

4. Les muscles moteurs de la respiration

Les muscles sont les générateurs des mouvements du tronc, ce que précisément nous voulons étudier.

Les muscles sont étudiés ici au travers de leurs compositions fibreuses, leurs propriétés mécaniques et leurs propriétés électriques. Les données anatomiques (et par voie de conséquence géométriques) décrites au paragraphe 1.4 ne seront pas ré-expliquées ici. Nous verrons que peu d'informations individuelles quantitativement précises sont disponibles. Malgré tout, nous essayerons de tirer parti de ces informations pour accroître notre connaissance du système et mettre en évidence les caractéristiques fondamentales à inclure dans la modélisation.

Les muscles participant à la respiration ne sont pas fondamentalement différents des autres muscles du squelette des mammifères. Dans ce paragraphe, nous ne décrirons pas la physiologie musculaire en détail. En particulier nous ne décrirons pas la mécanique, la biochimie complexe, ni les interactions moléculaires du processus de contraction (pour une approche simplifiée se référer à (Lombardo, 1996) et (Chen, 1992)).

4.1 Concepts fondamentaux

Pour donner une vue d'ensemble nous rappelons ici quelques concepts fondamentaux des muscles et de leurs dynamiques : les fibres constituantes, les propriétés mécaniques et les propriétés électriques.

4.1.1 Les fibres

Il existe trois types de fibres pour les muscles du squelette distinguées par leurs propriétés histochimiques et morphologiques, mais surtout par leur temps de contraction (Sharp et Hyatt, 1986) :

- Les fibres à contraction rapide, oxydante et glycolytique : Fast-twitch, Oxidative, Glycolytic (FOG).
- Les fibres à contraction rapide, glycolytique, mais non oxydante : Fast-twitch, Glycolytic (FG).
- Les fibres à contraction lente, oxydante : Slow-twitch, Oxydative (SO).

Généralement, on regroupe les deux premières, appelées aussi fibres rouges et fibres blanches, alors que les fibres lentes sont dénommées fibres intermédiaires. Les fibres rapides (FOG et FG) ont un temps de contraction court (les contractions sont isométriques) et une activité importante. Les FOG sont moins résistantes à la fatigue que les FG. Les SO ont un temps de contraction plus long et une activité plutôt réduite, leur résistance à la fatigue étant nettement plus importante. Le diamètre et la taille des fibres sont petits pour les FOG, moyens pour les SO et grands pour les FG.

Les FG sont utilisées pour les activités courtes, rapides et puissantes où la résistance à la fatigue et l'endurance ne sont pas nécessaires. Les FOG permettent une activité phasique soutenue où la résistance à la fatigue est désirable. Les SO sont lentes mais économiques, elles sont plutôt utilisées pour les activités toniques soutenues (maintien de la posture) où la résistance à la fatigue est primordiale. La plupart des muscles des mammifères contiennent un mélange des trois fibres. Le pourcentage de chacune des fibres dans un muscle nous donne

une indication sur l'utilisation de ce muscle. L'évolution de ce pourcentage en fonction de l'âge ou de pathologies est aussi une indication importante. Par exemple, pour un humain adulte moyen (d'environ 70Kg), les pourcentages des fibres diaphragmatiques sont de 21% de FOG, 24% de FG et 55% de SO. Les fibres ont un diamètre moyen de $34\mu\text{m}$. Pour donner une indication sur son activité, dans une respiration de repos, le diaphragme se contracte environ 15 fois par minute. Le diaphragme est donc un muscle d'activité régulière.

Si on classe les fibres selon leurs propriétés mécaniques on ne distingue que deux groupes : les fibres à contraction rapide (FOG et FG), constituantes essentielles des muscles rapides, et les fibres à contraction lente (SO), pour les muscles lents (remarquons que la constitution d'un muscle peut évoluer au cours du temps).

4.1.2 Propriétés mécaniques

Le modèle de muscle fondamental, le plus reconnu aujourd'hui, est le modèle de Hill (Figure 8 et (Sharp et Hyatt, 1986), (Chen, 1992)).

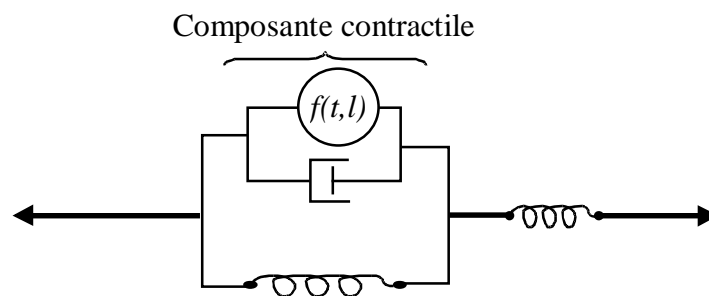


Figure 8. Modèle du muscle de Hill, constitué de trois composants en interactions : une composante contractile (fonction du temps et de la longueur des fibres), une composante élastique en série (représentant essentiellement l'effet du tendon aux points d'attache du muscle), et une composante élastique parallèle (passive, représentant l'élasticité des fibres).

Un autre modèle de muscle (considéré comme un raffinement du modèle de Hill) est le modèle Zajac. Ce dernier reflète mieux les non-linéarités dues aux glissements des filaments. Aux paramètres existant dans le modèle de Hill, le modèle de Zajac ajoute l'angle entre les fibres et les tendons auxquels elles sont rattachées.

Ces deux modèles nous donnent un moyen de calculer la force à générer à partir de la vitesse de contraction désirée. Cependant l'implication de la géométrie musculaire n'est que partiellement prise en compte dans ces modèles. Celle-ci est importante notamment pour la validation du modèle (forme et action sur les structures liées), mais aussi pour sa validité intrinsèque (contractilité correcte).

La géométrie est d'autant plus importante si l'on observe les courbes de tension/longueur des muscles (comme indiquée à la Figure 9).

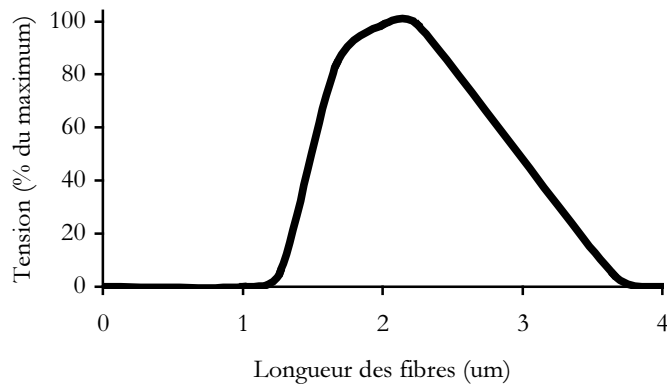


Figure 9. Courbe de tension/longueur d'un muscle de grenouille. La longueur des sarcomères (cellules contractiles des muscles) est donnée en μm (Sharp et Hyatt, 1986). Il y a un optimum de raccourcissement, au-delà la tension générée diminue.

Enfin, une dernière indication utile pour modéliser le comportement mécanique des muscles est leurs poids : leur densité moyenne est de 1,06 g/ml.

4.1.3 Propriétés électriques

On mesure les propriétés électriques des muscles grâce aux électro-myographies (EMG). Des électrodes sont disposées dans le muscle (aiguilles) ou à la surface de la peau. Elles reflètent la propagation du potentiel d'action du muscle due au flux ionique (généralement des ions Sodium) à travers les membranes des sarcomères.

La vitesse moyenne de propagation de ce signal à travers les fibres varie entre 2 et 6m/s (Sharp et Hyatt, 1986). Cette vitesse nous donne une indication fine sur les retards à introduire dans l'activation des forces. Dans un modèle élémentaire elle n'est pas nécessaire.

4.2 Propriétés individuelles

Examinons les propriétés élémentaires de chacun des muscles du système ventilatoire (voir (Sharp et Hyatt, 1986)).

4.2.1 Considérations générales

En 1963, Fenn met en évidence que la tâche mécanique des muscles respiratoires diffère de celle des muscles des autres membres car ils doivent lutter contre l'élasticité et la résistance (alors que les autres muscles luttent plutôt contre l'inertie). De plus les muscles respiratoires doivent être activés de manière répétée pour accomplir des actions de contraction et de génération de tension soutenues (alors que les autres muscles ont plutôt des activités brusques).

En ce qui concerne les facteurs géométriques, il faut en plus de l'allongement et du raccourcissement des fibres, considérer une autre influence plus globale. Par exemple,

L'augmentation du volume pulmonaire n'induit pas seulement un raccourcissement des fibres du diaphragme, mais aplatit et abaisse le diaphragme ce qui peut augmenter son rayon de courbure (particulièrement dans les cas de pathologie d'hyper-inflation pulmonaire). L'augmentation du rayon de courbure du diaphragme entraîne la réduction de la pression trans-diaphragmatique, déplaçant ainsi les paramètres mécaniques du système.

Pour la commande musculaire, des études chez l'homme, le chien et le chat suggèrent que la contraction volontaire maximale des muscles inspiratoires et expiratoires n'est pas contrainte par des mécanismes de contrôle neuraux, mais est principalement déterminée par les propriétés contractiles intrinsèques des muscles respiratoires.

Ces remarques établies, nous pouvons examiner les propriétés des principaux muscles respiratoires.

4.2.2 Diaphragme

Le diaphragme est composé comme on l'a vu de 45% de fibres rapides et 55% de fibres lentes (avec une prédominance de fibre SO dans la zone œsophagienne, c'est-à-dire la zone autour du cardia, le sphincter par lequel passe l'œsophage).

Les caractéristiques mécaniques sont donc intermédiaires entre les fibres lentes et les fibres rapides.

Lorsque le volume pulmonaire est faible, la courbe et le dôme du diaphragme sont plus importants. Cela optimise l'action du diaphragme non seulement parce que les fibres musculaires sont plus étirées, mais aussi parce que le faible rayon de courbure du diaphragme donne naissance à une pression trans-diaphragmatique plus importante.

Les études des EMG indiquent clairement que la partie crurale du diaphragme s'active immédiatement quand la commande nerveuse est envoyée, alors que la partie costale est recrutée petit à petit.

4.2.3 Muscles intercostaux

Il n'y a pas de différences de constitution notoires entre les fibres des intercostaux internes et externes. Elles sont pour la plupart des fibres SO (pour environ 65% des muscles).

Là encore, les propriétés musculaires sont intermédiaires entre fibres lentes et fibres rapides.

Par contre, aucune donnée n'est disponible pour décrire les courbes tension/longueur ou force/vitesse de contraction.

4.2.4 Sternomastoïdiens et scalènes

À notre connaissance, aucune publication n'est disponible sur le type de fibre composant les muscles scalènes.

Les sternomastoïdiens sont composés de 65% de fibres rapides et de 35% de fibres lentes. La dynamique est donc plus rapide pour ces muscles que pour le diaphragme ou les intercostaux. Cependant, là encore, aucune donnée n'est disponible pour caractériser les relations longueur/tension ou force/vitesse de contraction.

4.2.5 Abdominaux

Ils sont constitués d'environ 46% de fibres lentes et 54% de fibres rapides (l'oblique interne étant plutôt un muscle rapide, alors que l'oblique externe est un muscle lent). Cependant, les données disponibles sont peu précises car faites sur des expériences peu nombreuses.

Ici encore, à notre connaissance, aucune publication sur les courbes de force/vitesse ni sur les courbes longueur/tension n'est disponible.

4.2.6 Autres muscles

Aucune donnée n'a été publiée sur les propriétés mécaniques des muscles accessoires du système respiratoire (sauf sur le pectoral mineur qui est considéré comme un muscle lent). La composition des fibres est très peu connue en détail (sauf pour le pectoral majeur : 43% de fibres lentes et 57% de fibres rapides).

On voit ici que beaucoup de données détaillées manquent aux physiologistes et aux modélisateurs pour établir de manière précise les propriétés des différents muscles du système respiratoire. Cela entraîne deux conséquences :

- l'aspect géométrique étant le plus connu, c'est là que se portera notre attention,
- faute de moyens de comparaison et de vérification précis, les muscles ne pourront pas être modélisés de manière fine. C'est l'action globale des muscles qui guidera la modélisation. Nous allons voir maintenant quelle est cette action.

4.3 Action des muscles respiratoires

La mécanique respiratoire est un problème qui nécessite d'une part la connaissance détaillée d'un anatomiste classique et d'autre par la compréhension analytique de l'ingénieur.

W.O. Fenn, cité dans (De Troyer et Loring, 1986).

Comme on l'a vu, les muscles respiratoires sont conventionnellement divisés en trois groupes :

- le diaphragme,
- les muscles de la cage thoracique (intercostaux et accessoires),
- les muscles abdominaux.

Nous allons examiner l'action de ces trois groupes, voir (De Troyer et Loring, 1986).

4.3.1 Diaphragme

Le diaphragme ressemble à un cylindre elliptique coiffé par un dôme. Une première hypothèse est que la portion cylindrique ne change pas de forme au cours de la contraction, c'est la longueur axiale qui diminue pendant que le dôme descend relativement par rapport à ses insertions costales. La variation de forme la plus importante du diaphragme (celle entraînant la plus forte variation de volume) est le déplacement axial du dôme (déplacement similaire à celui d'un piston). Pendant des inspirations normales, la longueur axiale diminue de 2 cm.

Du point de vue historique, le diaphragme est considéré depuis longtemps comme l'élément moteur le plus important du système respiratoire et donc de la vie (« *la vie de l'esprit se situe dans le diaphragme* », attribué à Homère). Pour les Grecs, le diaphragme était en effet une structure purement métaphysique participant à l'âme et au processus de la pensée ! Galen, outre son approche plus scientifique du diaphragme (voir paragraphe 1.1), montre que les nerfs phréniques actionnent le diaphragme. Duchenne (1806-1875) découvre quant à lui l'importance de l'interaction entre le diaphragme et l'abdomen (« *l'expérience montre que plus la résistance au mouvement de contraction du diaphragme est importante, plus l'élevation des côtes est importante* » et « *grâce à la particularité de sa forme (les viscères abdominaux étant insérés à l'intérieur du dôme), la courbe naturelle du diaphragme est préservée lorsqu'il presse sur les viscères, permettant au diaphragme d'agir sur la cage thoracique dans une direction presque verticale et, par conséquent, de l'élever vers le haut* », cité par (De Troyer et Loring, 1986)).

Lorsque le diaphragme se contracte, la tension de ses fibres augmente causant un déplacement du dôme vers le bas. Les conséquences sont les suivantes :

- la pression intra-thoracique diminue et, en fonction de l'ouverture des voies aériennes, le volume pulmonaire augmente ou la pression alvéolaire diminue,
- la pression abdominale augmente et l'abdomen se déplace vers le bas et vers l'extérieur,
- la cage thoracique est déplacée.

Quelques remarques supplémentaires sont nécessaires :

- Influence de la posture : bien que le diaphragme déplace plus efficacement la cage thoracique lorsque le sujet est debout, il provoque, seul, des distorsions de la cage thoracique également lorsque le sujet est allongé.
- Quand les muscles de la cage thoracique ne fonctionnent pas, la contraction du diaphragme entraîne une diminution du volume de la partie haute de la cage thoracique.
- Plus l'action du diaphragme sur la cage thoracique diminue plus le volume pulmonaire est important.
- On considère aujourd'hui que le diaphragme est en fait constitué de deux muscles. La partie costale et la partie crurale agissent différemment sur la cage thoracique. La partie costale soulève le bas de la cage thoracique alors que la partie crurale ne fait rien (et ce même lorsque l'abdomen est ouvert). Cela provient de l'insertion des parties crurales et costales du diaphragme (seule la partie costale est insérée sur la cage thoracique). Cette considération explique de nombreux phénomènes générés en faisant varier la pression abdominale. Cependant, on ne sait toujours pas si ces deux parties *peuvent* se contracter séparément.

L'hypothèse simplificatrice émise précédemment à propos de la faible variation de forme du diaphragme au cours du cycle respiratoire semble aujourd'hui contestée. Dans (Petrol et al., 1990), deux modèles de contraction du diaphragme sont comparés avec des données réelles (obtenues par fluoroscopie à rayons X), voir Figure 10. Le but de cette modélisation est d'obtenir, à terme, la forme du diaphragme à tout moment du cycle respiratoire en n'utilisant que des données géométriques externes. Dans le premier modèle la contraction diaphragmatique est traitée comme si le diaphragme était un simple piston dans un cylindre (hypothèse de départ). Le deuxième modèle considère en plus un changement de configuration de la cage thoracique et de la forme du diaphragme. Ces travaux montrent

clairement que ce dernier modèle est plus précis (11% d'erreur par rapport aux données réelles) que le premier (30% d'erreur). Le changement de forme du diaphragme est donc un phénomène important à prendre en compte.

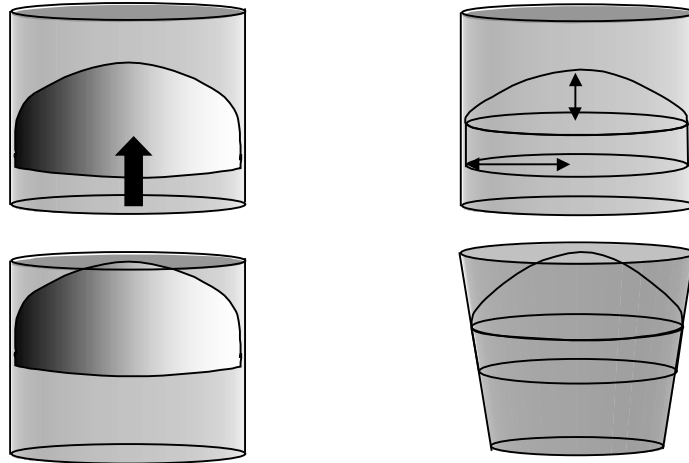


Figure 10. Deux modèles de variation de forme du diaphragme. À gauche le modèle simplificateur (le diaphragme se comporte comme un piston dans un cylindre). À droite le modèle de (Petrol et al., 1990), le diaphragme est composé d'un élément hémisphéroïde et d'un élément cylindrique tous deux déformables. En haut les modèles sont représentés en fin de la phase d'inspiration, en bas en fin de la phase d'expiration.

D'autre part, on doit aussi considérer la connexion du diaphragme avec les autres muscles du système respiratoire. La Figure 11 montre la nature série et parallèle de l'action musculaire du diaphragme sur le système ventilatoire.

L'étude du diaphragme est particulièrement complexe en raison de sa situation interne (donc difficile d'accès) et non proprioceptive (il ne peut pas communiquer des sensations indicatives de ses mouvements précis et de sa position exacte à l'intérieur du thorax).

4.3.2 Cage thoracique

Les muscles de la cage thoracique participant à la ventilation sont essentiellement les muscles intercostaux, les scalènes et les sternomastoïdiens.

Les actions des muscles intercostaux ont longtemps été une source de controverse dans l'histoire de la médecine et soulèvent encore de larges divergences d'opinion. On leur a successivement imputé une action expiratoire, inspiratoire, inspiratoire et expiratoire mais simultanée, puis inspiratoire et expiratoire mais en fonction de l'espace intercostal. On les considère aujourd'hui finalement comme régulateur de la tension intercostale.

Cette controverse est essentiellement due au fait qu'il est difficile d'étudier l'action séparée des muscles intercostaux internes et externes.

Les connaissances dans ce domaine sont encore insuffisantes. Par exemple on sait, par de nombreuses expériences, que les muscles intercostaux augmentent le volume thoracique, mais on ne sait pas exactement quels muscles sont utilisés pour cette action.

L'importance de ces muscles n'a jamais pu être bien étudiée, il se pourrait aussi que leur rôle majeur soit de transmettre le mouvement des côtes les plus basses aux côtes les plus hautes (mais il se peut aussi qu'ils n'agissent principalement que sur la posture).

Comme on l'a vu au paragraphe 4.2, les muscles scalènes et les sternomastoidiens ont une action mécanique très mal connue. Les seules données disponibles proviennent de sujets paralysés. On sait simplement que les scalènes ne sont utilisés que pour des ventilations importantes (et peu ou pas du tout pour une respiration de repos). Ils jouent surtout un rôle dans le déplacement vertical du sternum.

4.3.3 Abdomen

Les muscles de la paroi antéro-latérale ont un rôle primordial dans les mouvements du tronc (flexion et rotation). L'action principale de ces muscles durant la respiration concerne la compression du contenu abdominal. Les forces développées par les muscles abdominaux dépendent de trois paramètres :

- la tension développée dans les fibres musculaires,
- l'orientation des fibres,
- la distance entre les sites d'insertion musculaires sur les côtes et l'axe de rotation des côtes.

On les considère comme les muscles principaux de l'expiration pour deux raisons :

- la contraction des muscles de l'abdomen augmente la pression abdominale, déplaçant le diaphragme vers le haut.
- Leurs attachements anatomiques tendent à abaisser les côtes inférieures et donc à déplacer la cage thoracique vers le bas.

On peut aussi considérer une action inspiratoire accessoire de ces muscles : la pression interne étant plus importante les fibres musculaires du diaphragme et des muscles de la cage thoracique sont plus allongées au début de l'inspiration. Ces fibres sont alors déplacées sur une partie de la courbe tension/longueur plus avantageuse en terme de rendement.

En conclusion, les muscles abdominaux sont organisés et gérés de façon à améliorer les effets du diaphragme (générateur de pression et soulèvement de la partie basse de la cage thoracique).

4.3.4 Modèle mécanique de l'action musculaire

Le modèle présenté à la Figure 11 est du à P.T. Macklem, en 1978. Il montre la décomposition du diaphragme en deux muscles agissant différemment. Ce modèle est évidemment simplificateur car il décrit la cage thoracique comme une seule entité rigide (à un seul degré de liberté), et non pas comme une structure déformable. Cependant il caractérise bien l'interaction globale des muscles de la respiration.

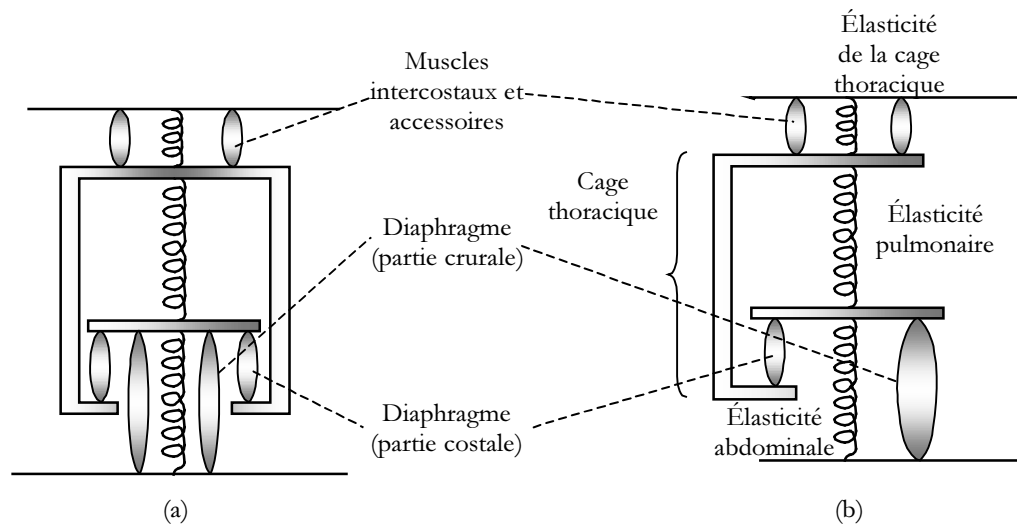


Figure 11. Modèle mécanique de l'action des muscles inspiratoires (de face et de profil) montrant les interactions relatives aux points d'encrages des différents muscles. L'élasticité des différentes structures est modélisée à l'aide de ressorts. Vue de face (a) et de profil (b).

Grâce à ces observations plus qualitatives que quantitatives, nous pouvons déterminer les éléments primordiaux à mettre en place dans notre modèle. C'est ce que nous allons voir en conclusion de ce chapitre.

5. Conclusion

On a vu dans ce chapitre comment fonctionne la respiration. Nous avons vu également que des paramètres intrinsèques précis des différents composants étaient difficiles à mesurer (et donc à trouver dans la littérature). Notre examen des différentes propriétés et relations des composants du tronc au cours de la respiration nous a amené à faire plusieurs constats que nous pouvons résumer par le Tableau 1.

Élément	Caractère	Déformation	Données
Cage thoracique	Éléments rigides articulés	Rotation, déformation des articulations (cartilages)	Δ volume, position extérieure
Diaphragme	Élément musculaire	Contraction/relaxation, position de repos	Force du muscle (en fonction de la pression trans-diaphragmatique)
Abdomen	Passif (dans un premier temps), ou paroi musculaire	Incompressible	Position des parois
Poumons	Fortement élastique	Occupent tout le volume déterminé par la cage thoracique	Δ volume, Δ pression

Tableau 1. Propriétés des différents éléments du tronc à prendre en compte dans la modélisation.

Au vu de cet état de l'art, les seules informations vraiment disponibles sur les propriétés des éléments du système ventilatoire sont plus qualitatives, alors que quantitativement on dispose essentiellement de données spatiales et géométriques (mesure de volume et de déplacement). Ce qui nous amène à faire quatre constats :

- il n'existe pas aujourd'hui de modèle mécanique et géométrique complet du système respiratoire ;
- notre modèle informatique du phénomène s'attachera principalement à retranscrire les mouvements des différentes structures du tronc ;
- il devra prendre en compte en priorité les paramètres géométriques ;
- l'objectif de la modélisation sera aussi de fournir un outil de compréhension et de vérification des hypothèses.

Deux types de connaissances sont à notre disposition : la mécanique (nous avons mis en évidence les propriétés nécessaires au modèle) et la géométrie (données anatomiques et imagerie médicale classique). Les conséquences du phénomène sont visibles par l'intermédiaire des déplacements et des déformations externes des composants du système ventilatoire, c'est pourquoi on s'oriente vers une représentation graphique du système ventilatoire.

Nous allons voir dans le chapitre suivant comment établir notre modélisation informatique du phénomène sur les bases de la littérature d'infographie (modélisation physique tridimensionnelle).

Chapitre 2

Simulation physique en infographie

Les théories qui résultent [de la mécanique] doivent permettre la prédiction au laboratoire des phénomènes que l'on observe par ailleurs. Il demeure par conséquent délicat de concevoir l'espace du physicien autrement qu'un espace d'observables, et par suite, géométrique.

Max Bausset (Bausset, 1982).

Un des buts de l'infographie est la synthèse d'images (représentation d'objets ou de scènes sur un écran d'ordinateur). Les objets affichés peuvent être réels ou virtuels, naturels ou artificiels. Le niveau de réalisme visuel des images de synthèse a toujours été une notion primordiale en infographie. Le vecteur de ce réalisme est soit l'image en elle-même (on cherche à obtenir des images les plus proches possibles d'images réelles ou de photographies de scènes naturelles), soit le comportement des objets (on cherche alors à représenter des objets dont les mouvements, les déformations et les interactions paraissent réels). L'ordinateur ayant des ressources limitées (aussi bien en calcul qu'en affichage), on doit se contenter aujourd'hui d'un compromis entre ces deux types de réalisme.

Dans le cadre de notre projet, notre principal objectif est d'obtenir une simulation la plus réaliste possible du comportement du tronc pendant la respiration.

Historiquement l'animation par ordinateur n'a tout d'abord été qu'une transposition informatique des techniques du dessin animé (utilisation de l'ordinateur pour calculer une interpolation entre des positions clés). Ces techniques ont rapidement évolué en parallèle avec le développement de la puissance de calcul des ordinateurs. La simulation d'objets solides et articulés a tout d'abord eu la faveur des informaticiens (la forme étant invariante, la simulation en est facilitée). Diverses techniques peuvent être utilisées pour calculer leur déplacement au cours d'une animation, notamment les lois issues de la physique des solides. Une part plus grande a été laissée aux ordinateurs, libérant le graphiste de tâches monotones et laissant une plus grande part à la création.

Le véritable défi informatique est la simulation d'objets déformables. Les modèles générateurs sont fondés à la fois sur :

- une description géométrique des objets (forme, position) et
- une description des lois régissant la déformation (souvent issues de lois physiques).

Ils sont très souvent appelés abusivement modèles physiques bien qu'ils n'utilisent que des simplifications de certaines lois physiques. En dix ans les modèles générateurs ont montré leur efficacité pour engendrer des mouvements complexes à partir d'une base génératrice simple. Le réalisme de ces mouvements et de ces déformations provient d'une bonne utilisation des lois d'animation. C'est pourquoi les lois de la physique ont été utilisées. Le problème fondamental provient de la nature discrète des calculateurs : on doit discrétiser les objets et les lois d'évolution (d'une manière ou d'une autre). De plus la complexité des mouvements possibles a vite fait apparaître le besoin de contraindre ces modèles (Platt et Barr, 1988).

On a vu au chapitre précédent que l'objectif de cette thèse se ramenait à modéliser et animer une image tridimensionnelle du tronc et de ses mouvements. Nous avons décomposé le tronc en structures essentielles chacune ayant un comportement propre.

Notre modèle sera un modèle générateur (basé sur la physique). Le présent chapitre introduit les différentes notions et idées correspondantes déjà développées dans la littérature d'informatique graphique. Ce chapitre ne présente pas une vision originale de la littérature, mais essaye de distinguer les idées et les concepts importants à retenir pour construire notre modèle. D'autres études bibliographiques récentes plus approfondies dans des domaines particuliers sont disponibles : synthèse d'images naturelles et techniques de rendu (Chaudy, 1997), théorie de l'élasticité (Ciarlet, 1985), (Bainville, 1996) et (Gascuel, 1990), objets déformables (Jimenez, 1993), (Gascuel et Puech, 1994), simulation physique (Joukhadar, 1997). (Foley et al., 1990) et (Watt et Watt, 1992) sont d'autres ouvrages de référence généraux dans le domaine de l'infographie, (Bausset, 1982) et (Larralde, 1986) dans le domaine de la dynamique.

Dans ce chapitre nous examinerons tout d'abord les lois physiques et les calculs nécessaires à la simulation d'objets solides, puis nous présenterons les différentes options possibles pour simuler des objets déformables. Nous verrons ensuite comment contraindre les modèles générateurs. Nous verrons enfin vers où se porte la problématique actuelle.

Notation : dans la suite de tout ce document, les notations mathématiques suivantes sont adoptées :

Variables		Opérations vectorielles		Pour \mathbf{x} fonction du temps	
a	Scalaire	$\langle \mathbf{u}, \mathbf{v} \rangle$	produit scalaire de \mathbf{u} et \mathbf{v}	$\dot{\mathbf{x}}$	dérivée première de \mathbf{x} par rapport au temps : $\dot{\mathbf{x}} = \frac{\partial \mathbf{x}}{\partial t}$
\mathbf{u}	Vecteur	$\mathbf{u} \times \mathbf{v}$	produit vectoriel de \mathbf{u} et \mathbf{v}		$\ddot{\mathbf{x}}$
\mathbf{M}	Matrice	\mathbf{uv}	vecteur $\mathbf{v} - \mathbf{u}$		
f	Fonction				

1. Principes généraux de la mécanique des corps solides et méthodes d'intégration

Dans ce premier paragraphe nous allons poser les bases de la modélisation mécanique et dynamique. C'est très souvent sur cette base que s'appuient les modèles générateurs que nous verrons au paragraphe 2.

Nous ferons tout d'abord un rappel de la mécanique du point, de celle des objets rigides (voir (Bausset, 1982) et (Larralde, 1986)), puis de la représentation de l'orientation des objets rigides. Le problème commun qui nous permettra de lier le présent paragraphe et le paragraphe 2 est celui de la méthode d'intégration. Nous l'examinerons au paragraphe 1.4.

1.1 Dynamique du point

La première description mécanique de l'univers physique peut se faire à l'aide de deux concepts : celui de force génératrice de mouvement et celui de point matériel.

Max Bausset (Bausset, 1982).

On considère un repère euclidien \mathfrak{R} de l'espace dans lequel on associe une position \mathbf{x} (de dimension 3) et une masse m à un point matériel. La loi d'évolution observée est le changement de position de ce point matériel dans \mathfrak{R} . On considère pour cela les torseurs cinématique et dynamique qui sont respectivement les torseurs associés à la quantité de mouvement et à la quantité d'accélération.

On ne s'intéresse ici qu'au torseur dynamique $A_{\mathbf{x}/\mathfrak{R}}$ dont les éléments de réduction en un point quelconque \mathbf{p} sont :

$$A_{\mathbf{x}/\mathfrak{R}\mathbf{p}} = \begin{cases} \mathbf{S}(A_{\mathbf{x}/\mathfrak{R}}) = m\ddot{\mathbf{x}}_{\mathbf{x}/\mathfrak{R}} \\ \mathbf{M}_{\mathbf{p}}(A_{\mathbf{x}/\mathfrak{R}}) = \mathbf{p}\mathbf{x} \times m\ddot{\mathbf{x}}_{\mathbf{x}/\mathfrak{R}} = m \cdot \mathbf{p}\mathbf{x} \times \ddot{\mathbf{x}}_{\mathbf{x}/\mathfrak{R}} \end{cases}$$

Pour simplifier l'écriture on considère \mathfrak{R} comme le repère par défaut, et le point de réduction au centre du repère. Par la suite, ils ne sont pas précisés dans les équations. On obtient plus simplement :

$$\begin{aligned} \mathbf{S} &= m\ddot{\mathbf{x}} \text{ (c'est la quantité d'accélération) et} \\ \mathbf{M} &= \mathbf{x} \times m\ddot{\mathbf{x}}, \text{ (c'est le moment dynamique).} \end{aligned}$$

Si l'on soumet ce point matériel à un ensemble de n forces \mathbf{f}_i , on peut alors énoncer la relation fondamentale de la dynamique (ou loi de Newton) :

$$\mathbf{S} = \sum_{i=1}^n \mathbf{f}_i = m\ddot{\mathbf{x}}.$$

Pour un système non dissipatif, on peut aussi exprimer la dynamique d'un point matériel sous forme énergétique. Cette expression sera utilisée plus tard dans certains modèles physiques.

Sous forme énergétique, on a :

$$\frac{\partial E_{mec}}{\partial t} = \frac{\partial (E_{cin} + E_{pot})}{\partial t} = 0,$$

où $E_{cin} = \frac{1}{2} m \dot{\mathbf{x}}^2$ est l'énergie cinétique du point matériel, et $\Delta E_{pot} = -\int_{\mathbf{x}}^{\mathbf{x}+\Delta \mathbf{x}} \sum \mathbf{f}_i \cdot \delta \mathbf{x}$ est l'énergie potentielle du point (travail des forces extérieures). Dans le cas de deux masses reliées par un ressort, par exemple, l'énergie cinétique du système provient du mouvement des masses, l'énergie potentielle de la tension stockée dans le ressort.

Un autre principe sera utilisé dans certains modèles générateurs : le principe des travaux virtuels. On définit le travail virtuel \mathbf{W} d'une force \mathbf{f}_i par :

$$\delta \mathbf{W} = \mathbf{f}_i \cdot \delta \mathbf{x},$$

où $\delta \mathbf{x}$ représente le déplacement virtuel.

La variation de l'énergie potentielle de \mathbf{f}_i (appliquée en \mathbf{x}) par rapport au temps est appelée le travail de cette force. Dans un système en équilibre, la variation de l'énergie cinétique est nulle. Dans ce cas, le théorème des travaux virtuels (ou principe des travaux virtuels d'Alembert) indique que, pour tout déplacement virtuel, la somme algébrique des travaux virtuels des forces extérieures \mathbf{W}^{ext} et des travaux virtuels des forces intérieures \mathbf{W}^{int} est nulle :

$$\mathbf{W}^{int} + \mathbf{W}^{ext} = \mathbf{0}.$$

En synthèse d'images on cherche à obtenir l'évolution des trajectoires et des déformations des objets d'une scène. À cause du type de fonctionnement de l'ordinateur, nous devons discrétiser cette évolution pour produire des images à un instant donné (en fait on s'applique à produire des séquences d'instantanés). La simulation informatique des mouvements basée sur la physique suivra toujours le schéma suivant :

- calcul du rendu et affichage de l'image,
- calcul des forces appliquées aux objets à partir de la position actuelle,
- discrétisation des équations de la dynamique et calcul de la nouvelle position.

Ce schéma se distingue de celui utilisé dans les ouvrages de mécanique classique où l'on part généralement d'une observation de trajectoire et où l'on cherche les forces qui l'influencent. Ici l'ensemble des forces appliquées aux objets est connu.

Ainsi, connaissant l'ensemble des forces appliquées à un point matériel et un schéma d'intégration, on pourra décrire l'évolution de \mathbf{x} . Les méthodes et problèmes posés par l'intégration de la loi de Newton seront exposés au paragraphe 1.4.

1.2 Dynamique des objets rigides

Nous décrivons dans ce paragraphe les lois de la dynamique des objets rigides discrets (en précisant pour la plupart les formulations continues dont elles sont issues). En annexe 1, on trouvera les algorithmes permettant de simuler cette dynamique.

Dans la plupart des cas à traiter en simulation informatique, la forme de l'objet ne s'exprime pas analytiquement, on ne s'intéressera donc pas aux lois continues. Nous devons pouvoir

traiter une forme et une répartition de masse quelconque. De plus, nous ne considérons que le cas d'objets rigides de masse m discrétisés en un ensemble n de points matériels. Cette forme de discrétisation est la plus répandue en infographie du fait de l'utilisation d'un ordinateur dans les phases de calculs, de manipulation et d'affichage. Chaque point matériel appartenant à l'objet est caractérisé par une masse m_i et une position $\mathbf{x}_i = (x_i, y_i, z_i)$.

Comme il s'agit d'un objet rigide, le nombre de degrés de liberté est six (trois en rotation et trois en translation). La position du centre de gravité (barycentre des masses) de l'objet est notée \mathbf{x}_G . On la calcule de manière suivante :

$$\mathbf{x}_G = \frac{1}{m} \sum_{i=1}^n m_i \mathbf{x}_i \quad (\text{discrétisation de la formule continue : } \mathbf{x}_G = \frac{1}{m} \int \mathbf{x} dm, \text{ où } \mathbf{x}$$

décrit l'ensemble des points de l'objet).

Les trois degrés de liberté en rotation sont représentés classiquement par une matrice de rotation \mathbf{R} ou un quaternion \mathbf{q} (voir paragraphe 1.3 et annexe 2).

Pour caractériser le mouvement du solide, il faut aussi connaître sa matrice d'inertie \mathbf{I} (dont les éléments sont exprimés en $kg.m^2$), calculée par :

$$\mathbf{I} = \begin{bmatrix} \sum m_i (y_i^2 + z_i^2) & -\sum m_i x_i y_i & -\sum m_i x_i z_i \\ -\sum m_i y_i x_i & \sum m_i (x_i^2 + z_i^2) & -\sum m_i y_i z_i \\ -\sum m_i z_i x_i & -\sum m_i z_i y_i & \sum m_i (x_i^2 + y_i^2) \end{bmatrix},$$

qui provient de la discrétisation de la formule continue :

$$\mathbf{I} = \begin{bmatrix} \int (y^2 + z^2) dm & -\int xy dm & -\int xz dm \\ -\int yx dm & \int (x^2 + z^2) dm & -\int yz dm \\ -\int zx dm & -\int zy dm & \int (x^2 + y^2) dm \end{bmatrix}.$$

Pour un objet discrétisé en points matériels, le torseur dynamique au point de réduction \mathbf{x}_G s'exprime par :

$$\mathbf{S} = \sum_i m_i \ddot{\mathbf{x}}_i \quad (\text{somme des quantités d'accélération relatives à chacun des points}$$

du système) et

$$\mathbf{M} = \sum_i \mathbf{x}_G \mathbf{x}_i \times m_i \ddot{\mathbf{x}}_i \quad (\text{moment au centre des masses}).$$

Pour un solide les éléments du torseur dynamique s'expriment sous forme continue par :

$$\mathbf{S} = \int \ddot{\mathbf{x}} dm, \text{ et}$$

$$\mathbf{M} = \int \mathbf{x}_G \mathbf{x} \times \ddot{\mathbf{x}} dm.$$

Ensuite à chaque étape de la simulation, à partir des forces appliquées en chaque point matériel de discrétisation de l'objet considéré, on calcule la quantité d'accélération et le moment au centre de gravité :

$$\sum_i \mathbf{f}_i = m\ddot{\mathbf{x}}_G,$$

$$\mathbf{M} = \sum_i \mathbf{x}_G \mathbf{x}_i \times \mathbf{f}_i,$$

où $\mathbf{f}_i = \sum_j \mathbf{f}_j$ est la somme des forces \mathbf{f}_j qui s'appliquent en \mathbf{x}_i . En effet, le centre d'inertie

d'un objet se comporte comme une masse ponctuelle égale à m subissant l'ensemble des forces appliquées à ses points matériels.

On pose \mathbf{w} vitesse angulaire de l'objet. L'expression de la loi fondamentale de la dynamique entre la quantité de mouvement angulaire $\mathbf{L} = \mathbf{I}\mathbf{w}$ et le moment au centre d'inertie \mathbf{M} permet de calculer la nouvelle orientation :

$$\mathbf{M} = \frac{d\mathbf{L}}{dt} = \frac{d\mathbf{I}\mathbf{w}}{dt}.$$

Ainsi à partir de la position, de l'orientation et de la somme des forces à l'instant t , on peut calculer l'évolution de l'objet. On trouvera en annexe 1 l'ensemble de la formulation ainsi que les algorithmes permettant de mettre en œuvre les mouvements d'un objet rigide composé d'un ensemble de masses ponctuelles.

Nous allons maintenant regarder de plus près comment représenter l'orientation du solide.

1.3 Représentation de l'orientation des solides

Un objet solide a six degrés de liberté dont trois sont aisément représentés par \mathbf{x}_G (position du centre de gravité). Il existe trois façons de représenter les trois derniers degrés de liberté (l'orientation du solide) :

- Matrice orthogonale de rotation classique (équivalent à l'utilisation des angles d'Euler) : c'est en fait la composition de trois rotations (selon les trois axes). Cette solution pose deux problèmes : elle ne permet pas de décrire toutes les rotations (celles dont l'axe est contenu dans le plans $x = z$ et $x = -z$) et elle pose dans certains cas des problèmes de différenciation.
- Vecteur de rotation et angle de rotation (en utilisant la formule de Rodrigues). Il n'y a pas unicité de la représentation (le couple (\mathbf{v}, θ) représente la même rotation que le couple $(-\mathbf{v}, 2\pi - \theta)$).
- Quaternions : dans ce cas il n'y a pas de singularité. De plus, les manipulations sont plus performantes et minimisent l'erreur propagée lors des calculs successifs.

Nous avons utilisé cette dernière représentation, c'est pourquoi nous en détaillerons rapidement quelques aspects. On pourra trouver des études plus complètes dans (Shoemaker, 1985), (Gini et Gini, 1985). (Bainville, 1996) fait une étude beaucoup plus rigoureuse et détaillée sur la représentation des rotations.

Inventé en 1843 par Sir W.R. Hamilton, les quaternions n'ont été utilisés comme alternative aux angles d'Euler que depuis peu (Shoemaker, 1985). Les quaternions sont une généralisation des nombres complexes à trois parties imaginaires :

$$\begin{aligned} \mathbf{q} &= q_0 + \mathbf{i} \cdot q_1 + \mathbf{j} \cdot q_2 + \mathbf{k} \cdot q_3, \\ \text{avec } \langle \mathbf{i}, \mathbf{j} \rangle &= -\langle \mathbf{j}, \mathbf{i} \rangle = \mathbf{k}, \langle \mathbf{j}, \mathbf{k} \rangle = -\langle \mathbf{k}, \mathbf{j} \rangle = \mathbf{i}, \langle \mathbf{k}, \mathbf{i} \rangle = -\langle \mathbf{i}, \mathbf{k} \rangle = \mathbf{j}, \\ \text{et } \mathbf{i}^2 &= \mathbf{j}^2 = \mathbf{k}^2 = \langle \mathbf{i}, \langle \mathbf{j}, \mathbf{k} \rangle \rangle = -1. \end{aligned}$$

Une rotation d'axe unitaire \mathbf{v} et d'angle θ est représentée par le quaternion :

$$\mathbf{q} = \cos \frac{\theta}{2} + \mathbf{v} \sin \frac{\theta}{2} = \left(\cos \frac{\theta}{2}, x_v \sin \frac{\theta}{2}, y_v \sin \frac{\theta}{2}, z_v \sin \frac{\theta}{2} \right).$$

La composition de deux rotations s'obtient par multiplication des deux quaternions correspondant.

On trouvera dans la littérature citée plus haut et en annexe 2 tous les algorithmes de passage entre la matrice de rotation et le quaternion.

La stabilité numérique des quaternions est à la fois une conséquence du nombre d'opérations (moindre par rapport aux matrices) et du fait que les cosinus et sinus ne sont pas utilisés directement.

En fait, il nous faut conserver les deux représentations pour optimiser tous les calculs en prenant les avantages de chacune d'entre elles (voir annexe 1).

1.4 Méthode d'intégration

Pour une approche plus approfondie des méthodes d'intégration on se reportera à (Witkin et al, 1995) et (Press et al, 1988).

Dans toute simulation physique, nous sommes confrontés à des équations différentielles du type :

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t).$$

En résolvant cette équation différentielle ordinaire, nous pouvons produire des « instantanés » du système. À chaque moment de la simulation des changements dans les équations peuvent intervenir : addition d'une nouvelle force, accident de collision ou interaction avec le manipulateur. C'est pourquoi on ne peut utiliser une solution analytique. Nous devons discrétiser ces équations dans le temps. On part de la valeur de la variable $\mathbf{x}(t_0)$ (valeur initiale à $t = 0$) et de la fonction \mathbf{f} pour calculer la variation de \mathbf{x} au cours d'un intervalle de temps dt et ainsi obtenir la nouvelle valeur de \mathbf{x} : $\mathbf{x}(t_0 + dt)$.

On peut trouver dans la littérature classique plusieurs schémas d'intégration dont les plus connus sont : Euler, Newton-Cotes (voir annexe 1) et Runge-Kutta, tout trois font appel aux développements limités (série de Taylor) :

$$\begin{aligned} \mathbf{x}(t_0 + dt) &= \sum_{i=0}^n \frac{dt^i}{i!} \frac{\partial^i \mathbf{x}}{\partial t^i} + O\left(\frac{dt^{n+1}}{(n+1)!} \frac{\partial^{n+1} \mathbf{x}}{\partial t^{n+1}}\right) \\ &= \mathbf{x}(t_0) + dt \cdot \dot{\mathbf{x}}(t_0) + \frac{dt^2}{2!} \ddot{\mathbf{x}}(t_0) + \dots \end{aligned}$$

1.4.1 Schéma d'Euler :

On obtient le schéma d'Euler simplement en tronquant la série de Taylor au deuxième élément. L'erreur est donc en $O\left(\frac{dt^2}{2}\ddot{\mathbf{x}}(t_0)\right)$.

1.4.2 Schéma de Newton-Cotes

Cette fois-ci on utilise la moyenne entre $f(\mathbf{x}, t_0)$ et $f(\mathbf{x}, t_0 + dt)$ comme évaluation de la fonction f .

1.4.3 Schéma de Runge-Kutta

Pour cette méthode, on va de nouveau appliquer la série de Taylor, mais cette fois-ci à la fonction f . L'ordre de la méthode de Runge-Kutta est le nombre de fois où l'on va subdiviser et réévaluer f . Par exemple pour Runge-Kutta d'ordre 4, on a :

$$\begin{aligned} k_1 &= dt \cdot f(\mathbf{x}(t_0), t_0) \\ k_2 &= dt \cdot f\left(\mathbf{x}(t_0) + \frac{k_1}{2}, t_0 + \frac{dt}{2}\right) \\ k_3 &= dt \cdot f\left(\mathbf{x}(t_0) + \frac{k_2}{2}, t_0 + \frac{dt}{2}\right) \\ k_4 &= dt \cdot f(\mathbf{x}(t_0) + k_3, t_0 + dt) \\ \mathbf{x}(t_0 + dt) &= \mathbf{x}(t_0) + \frac{1}{6}k_1 + \frac{1}{3}k_2 + \frac{1}{3}k_3 + \frac{1}{6}k_4 \end{aligned}$$

Runge-Kutta nécessite donc plusieurs évaluations de la fonction f (dans notre cas cela veut dire plusieurs évaluations des forces appliquées sur l'objet).

Dans ces méthodes le pas de temps est déterminant (plus le pas de temps est petit plus l'erreur commise est petite). D'après le théorème de Shannon, pour éviter toute divergence d'un système il faut au moins échantillonner le pas de temps à valeur de deux fois la fréquence la plus haute du système.

L'erreur commise est difficilement évaluable au cours d'une simulation. C'est pourquoi (Joukhadar, 1997) propose de calculer l'état énergétique des objets et d'utiliser cette grandeur pour faire varier le pas de temps. Par principe, l'énergie mécanique d'un système physique (voir paragraphe 1.1) est constante pour un système isolé sans dissipation d'énergie. Le système physique n'est pas isolé (des forces extérieures agissent sur lui); pour cette raison l'auteur considère que le système est stable lorsque la variation de son énergie mécanique est inférieure à un certain seuil. Si ce seuil est franchi, le pas de temps est divisé par deux, si le seuil n'est pas franchi, il est multiplié par deux.

Ici encore, on doit faire face à un dilemme : plus la méthode est simple, plus le temps de calcul nécessaire est petit, mais plus l'erreur commise est grande (pour une analyse de l'erreur commise voir (Lombardo, 1996)). D'autre part, si le critère de choix du pas de temps est l'erreur (on souhaite par exemple que l'erreur soit inférieure à un certain seuil), les méthodes

d'intégration plus complexes peuvent permettre d'augmenter le pas de temps (et donc de réduire le nombre d'évaluation de la fonction).

2. Les modèles déformables

Les applications des modèles déformables en informatique sont très nombreuses : conception et modélisation d'objets (c'est d'ailleurs de ce domaine que sont issus les premiers outils de déformation de courbe et de surface), segmentation d'images (en médecine et en robotique notamment) et enfin simulation. C'est dans cette dernière catégorie que se situe notre travail.

Nous avons choisi de limiter l'étude des modèles déformables aux modèles générateurs car ce sont les seuls capables de répondre à notre besoin pour la simulation (en terme de réalisme visuel aussi bien que d'intégration dans le cadre des Gestes Médicaux Chirurgicaux Assistés par Ordinateur).

Au-delà des modèles déformables, nous devons examiner les solutions proposées de déformations élastiques. On trouvera dans (Watt et Watt, 1992) divers types d'autres déformations possibles notamment celles relatives à la modélisation d'objets tridimensionnels et à la Conception Assistée par Ordinateur, où le plus important n'est pas le comportement mais la facilité de contrôle (comme par exemple les déformations de formes libres (Sederberg et Parry, 1986)). Dans notre cas, il ne suffit pas que les objets puissent se déformer, mais il faut qu'ils aient une réelle propriété élastique. L'élasticité est l'aptitude d'un corps à reprendre sa forme d'origine (ou forme de repos) lorsqu'il a subi une déformation. En fait, il s'agit d'une façon particulière de restituer l'énergie fournie. Les déformations dues à l'élasticité sont donc des forces de réaction à la déformation et doivent être nulles dans le cas d'un déplacement rigide.

On parle d'hyper-élasticité lorsque les forces élastiques dérivent d'un potentiel de déformation.

L'implémentation informatique de l'élasticité doit être telle que :

- l'élasticité agit directement sur la forme de l'objet,
- la forme de repos de l'objet est stockée d'une manière ou d'une autre,
- l'élasticité est générée en utilisant un opérateur de différence entre la forme déformée et la forme de repos (dépendant fortement des deux premiers points).

Ainsi, l'ensemble des méthodes présentées ici diffèrent par la façon de manipuler la forme de l'objet, la façon d'agir sur celle-ci, et la façon de calculer la différence avec la forme de repos. Par la suite on s'attachera donc pour chaque méthode à préciser :

- le principe utilisé dans la loi d'évolution,
- la représentation de la forme des objets,
- la mémorisation de la forme de repos,
- l'opérateur de différence.

L'histoire de la simulation des corps déformables basés sur la physique en infographie a aujourd'hui dix ans. C'est en 1987 que D. Terzopoulos propose le premier modèle d'objets

déformables basé sur une discrétisation des lois de l'élasticité : (Terzopoulos et al., 1987). Depuis, bien d'autres techniques ont été utilisées pour simuler des objets déformables à partir d'une base de lois ou de règles, en tenant compte de l'interaction entre plusieurs objets ou l'utilisateurs.

On peut classer les modèles générateurs en deux grands groupes :

- les méthodes nodales : une phase de discrétisation en nœuds élémentaires intervient dans la simulation. Un nœud est la décomposition élémentaire auquel on attribue des caractéristiques (par exemple une masse et une position) et qui est manipulé par l'ordinateur.
- Les méthodes globales : les déformations sont calculées de manière globale (matrice de déformation globale, modes vibratoires).

C'est surtout la première approche qui nous intéresse puisqu'elle permet la modélisation d'objets non-homogène et facilite le contrôle local, l'interactivité et la décomposition.

Le paragraphe 2 décrit les principales approches des deux catégories. On ne parlera pas ici des problèmes posés par l'intégration temporelle des lois de déformations (vu au paragraphe 1.4).

2.1 Approches nodales

Par approche nodale on entend toute approche basée sur une discrétisation en nœuds à un moment ou à un autre de la chaîne de traitement. C'est l'approche la plus répandue en infographie. Toutes ces approches sont fortement connectées entre elles (bien que la forme et les buts semblent a priori souvent différents). Voici leurs propriétés communes :

- utilisation de lois de déformations (issues de simplification ou de discrétisation de lois physiques),
- utilisation d'un moteur d'intégration (permettant le calcul du mouvement d'un instant t à un instant $t + \Delta t$),
- discrétisation des objets en nœuds ou assemblage d'éléments individuels,
- application des lois de déformations au niveau des nœuds.

On distingue la discrétisation d'un maillage (de haut en bas, essentiellement méthodes des différences finies et des éléments finis) et l'assemblage d'éléments (de bas en haut, comme les réseaux de masses-ressorts, les systèmes de particules, ou les surfaces implicites).

2.1.1 Discrétisation d'objets

Ces méthodes reposent sur les théories de l'élasticité et utilisent les méthodes de discrétisation classiques. On distingue deux types de discrétisation :

- discrétisation par la méthode des différences finies et
- discrétisation par la méthode des éléments finis.

a) Discrétisation par la méthode des différences finies

La méthode est la suivante : on part de la formulation lagrangienne de l'équation d'évolution. La forme des objets est représentée de manière paramétrée (dans le temps et dans l'espace) :

\mathbf{x}_a^t est le vecteur position où \mathbf{a} est un vecteur (de dimension respectivement un, deux ou trois pour respectivement une courbe, une surface, un solide) contenant les valeurs du

paramètre de discrétisation de l'objet et t représente le temps. On utilise la méthode des différences finies pour effectuer les calculs.

La forme de repos est mémorisée grâce au tenseur métrique et au tenseur de courbure défini sur les nœuds de discrétisation : $\mathbf{x}_a^0 = \mathbf{x}_a^{t_0}$.

On considère de plus un modèle hyper-élastique dans lequel on dérive le potentiel de déformation pour obtenir les forces à appliquer. On formule l'énergie de déformation $\mathcal{E}(\mathbf{x})$ comme étant la distance entre la forme de l'objet au repos et la forme de l'objet déformé (on utilise toujours les tenseurs métriques et de courbure).

Si μ_a est la densité au point \mathbf{a} , γ_a l'amortissement au point \mathbf{a} , et $f(\mathbf{x}_a)$ les forces extérieures appliquées sur l'objet, l'équation d'évolution du point \mathbf{a} est donc la suivante (forme lagrangienne) :

$$\mu_a \ddot{\mathbf{x}}_a + \gamma_a \dot{\mathbf{x}}_a + \frac{\partial \mathcal{E}(\mathbf{x})}{\partial \mathbf{x}} = f(\mathbf{x}_a). \quad (1)$$

À chaque itération, on a toutes les informations pour calculer la réponse élastique de l'objet. L'équation ci-dessus est formulée pour chaque point \mathbf{a} (n équations si on a une discrétisation en n éléments). On rassemble les équations dans un système matriciel :

$$\mathbf{M}\ddot{\mathbf{x}} + \mathbf{C}\dot{\mathbf{x}} + \mathbf{K}(\mathbf{x})\mathbf{x} = \mathbf{f}, \quad (2)$$

où \mathbf{x} est le vecteur de dimension n contenant les positions des points \mathbf{a} , \mathbf{M} est la matrice diagonale de répartition des masses, \mathbf{C} la matrice d'amortissement, \mathbf{f} le vecteur des forces extérieures exercées sur les points \mathbf{a} de l'objet et $\mathbf{K}(\mathbf{x})$ la matrice d'élasticité (ou matrice de raideur). $\mathbf{K}(\mathbf{x})$ est la formulation discrète du gradient de l'énergie $\mathcal{E}(\mathbf{x})$. Ce système est ensuite discrétisé dans le temps ($\mathbf{K}(\mathbf{x})$ sera calculée à chaque itération).

En fait, cette méthode est strictement équivalente à un réseau de masses ponctuelles dont le maillage serait régulier en position de repos.

Outre le temps de résolution important, ce modèle ne permet pas, d'après les auteurs, de modéliser des objets très rigides à cause de sa sensibilité à la divergence.

Dans (Terzopoulos et Witkin, 1988), les auteurs proposent d'améliorer le temps de calcul et la stabilité du modèle précédent par une formulation hybride. L'objet déformable est décomposé en deux composantes : une composante de référence et une composante déformable. La composante de référence se comporte comme un objet rigide. Cela diminue considérablement le temps de calcul, mais oblige à simplifier fortement l'expression de la déformation. De plus, le mouvement de l'objet déformable est moins réaliste puisqu'il est approximé par le mouvement d'un corps rigide sous-jacent. Dans (Terzopoulos et Fleischer, 1988), on utilise toujours le même modèle mais la composante élastique transmet de l'énergie de déformation au composant de référence. Cela permet de modéliser la viscoélasticité, la plasticité et les fractures. Cependant la modélisation reste la même (l'objet est toujours connexe, la matrice $\mathbf{K}(\mathbf{x})$ est unique).

a) Discrétisation par la méthode des éléments finis

La méthode des éléments finis est une des méthodes de modélisation les plus utilisées dans l'ingénierie, notamment pour la modélisation de structure rigide ou en thermodynamique (pour la méthode des éléments finis voir (Bainville, 1996) et (Zienkiewicz et Taylor, 1991)).

Les objets sont discrétisés en volumes ou en mailles élémentaires e comportant un nombre connu de nœuds (le plus souvent huit ou neuf nœuds). À partir de la position des nœuds de e , des fonctions d'interpolation permettent de connaître la position de tous les points de e . Les équations énergétiques de déformation peuvent être appliquées aux nœuds, il en résulte un système dont la résolution fournit les déplacements des nœuds. On utilise pour cela le principe des travaux virtuels d'Alembert qui postule la conservation de l'énergie. Ce qui nous donne, dans un cadre dynamique (la méthode des éléments finis peut s'appliquer à d'autres cas), l'équation d'évolution classique :

$$\mathbf{M}\ddot{\mathbf{x}} + \mathbf{C}\dot{\mathbf{x}} + \mathbf{K}\mathbf{x} = \mathbf{f},$$

déjà vue au paragraphe précédent. Deux choix sont possibles : trouver la solution directe (implicite) à partir du système ou discrétiser l'équation dans le temps (utiliser un schéma explicite, type Euler ou Newton-Cotes). Même en admettant que l'amortissement dépend linéairement des masses des nœuds (c'est-à-dire que la matrice $\mathbf{C} = k\mathbf{M}$), la première solution reste lourde (il faut toujours inverser la matrice \mathbf{K}). Généralement on utilise la deuxième solution qui, bien que moins stable, reste simple à mettre en œuvre.

Les auteurs de (Gourret et al., 1989) utilisent cette méthode dans le cadre de l'infographie. Au départ il s'agit de modéliser une main qui serre une balle élastique.

L'une des idées principales est de ne considérer que les contacts prolongés entre les objets : le système est à l'équilibre à tout instant, l'évolution devient quasi-statique. On utilise donc le principe des travaux virtuels d'Alembert (le travail des forces externes appliquées à l'objet doit être égal au travail des forces internes). On décompose les forces externes en trois groupes : les forces globales (champs de forces, gravité), les forces de surfaces (pression), et les forces locales (collision, interaction locale). On ré-exprime donc l'égalité due au théorème des travaux virtuels :

$$W_{global} + W_{surface} + W_{locales} = W_{internes} \quad (3)$$

Il n'y a pas de dynamique. Pour modéliser une loi d'élasticité linéaire, on ne doit résoudre que :

$$\mathbf{K}\mathbf{x}_d = \mathbf{f}, \quad (4)$$

où \mathbf{K} est la matrice de raideur, \mathbf{x}_d le vecteur des déplacements par rapport à l'état initial, et \mathbf{f} le vecteur des forces appliquées à l'objet. On reconnaît ici l'équation d'un ressort. Dans (Gourret et al., 1989), les éléments e comportant huit nœuds, \mathbf{x}_d est un vecteur de taille $3n \cdot 8$, où n est le nombre d'éléments.

Pour modéliser la main, on utilise son squelette pour guider l'opération. À chaque instant, on détermine les nouvelles positions de la main (positions connues). Les points de la balle serrée en contact avec la main ont donc des positions connues. À partir de l'équation ci-dessus, les positions des points de la balle qui ne sont pas en contact avec la balle seront calculées. C'est pourquoi on va partitionner le vecteur \mathbf{x}_d en deux (les déplacements connus et ceux que l'on cherche).

Dans (Bainville, 1996), l'algorithme d'évolution minimise l'énergie élastique (en maintenant à chaque instant le système dans un minimum local à l'intérieur de l'espace non contraint).

En contrepartie de son réalisme, la méthode des éléments finis est connue pour nécessiter une grande puissance de calcul (bien que des simplifications soient prévues dans cette

méthode). La matrice de raideur nous oblige à ne considérer qu'un seul objet connexe sans changement de topologie. De plus on ne peut calculer la matrice de raideur que lorsque certains points sont fixes.

Nous verrons que (Chen, 1992) utilise aussi la méthode des éléments finis pour modéliser les muscles.

Dans (Cotin et al, 1996) les objets sont discrétisés en tétraèdres sur lesquels on applique la méthode des éléments finis. Cette méthode est ici appliquée à la déformation du foie par un instrument chirurgical.

Là encore le comportement de l'objet dépend fortement de sa discrétisation. Le problème est alors de trouver un compromis pour obtenir une discrétisation efficace mais réaliste.

2.1.2 Assemblage d'éléments

L'assemblage d'éléments consiste à utiliser un ensemble d'éléments ayant des caractéristiques propres. On assemble ces éléments pour former un objet. L'évolution de l'objet n'est considérée que par l'intermédiaire des éléments le composant. On n'utilisera pas le terme de particules pour éviter toute confusion (on réserve ce terme pour les modèles spécifiques de systèmes de particules).

Pour ces méthodes, chaque élément possède des caractéristiques individuelles (ce qui permet, contrairement aux approches de discrétisation, de modéliser des objets non-homogènes). À chaque itération de la simulation l'algorithme calcule la somme des forces appliquées à chaque élément.

Pour modéliser des comportements élastiques on utilise soit des relations élastiques linéaires entre les éléments (on relie les éléments par des ressorts), soit des lois de comportement (interaction entre éléments lorsqu'ils sont suffisamment proches). Une composante de viscosité peut compléter ces liaisons.

L'avantage de ces modèles provient de la simplicité de la mise en œuvre (aussi bien pour les lois d'évolution que pour l'affichage à l'aide d'une carte graphique standard). De nombreuses études ont été menées sur ces modèles afin d'en améliorer les possibilités (collision, changement de topologie).

On peut en distinguer trois formes principales : les réseaux explicites, les systèmes de particules, et les surfaces implicites.

a) Réseaux explicites

Dans les réseaux explicites chacun des éléments connaît son voisinage explicitement. Celui-ci peut varier au cours du temps si les fractures ou les déchirures sont modélisées. De nombreuses approches ont déjà été proposées. C'est certainement la méthode la plus intuitive pour former un objet à partir d'éléments. On confond souvent ces modèles avec le modèle classique « masse-ressort » duquel ils sont issus. On ne présentera ici que quelques références significatives.

Le modèle développé dans la présente thèse appartient à cette catégorie.

Modèle de Miller (Miller, 1988)

Le but est ici de simuler le mouvement de vers et de serpents. C'est donc un modèle très simple où l'animal est discrétisé en segment. À chaque segment on associe un point matériel. Des ressorts relient les cubes dans l'axe de l'animal et à travers les diagonales de chaque face.

Une composante d'amortissement fonction de la vitesse est ajoutée aux ressorts. On ajoute des forces externes comme la gravité. Les collisions avec le sol sont prises en compte.

On utilise le schéma d'intégration d'Euler pour calculer l'évolution en contractant les ressorts de manière cyclique.

CORDIS-ANIMA, le modèle de l'ACROE (Luciani et Cadoz, 1986)

Dans le modèle initial de l'ACROE, un objet est constitué de points matériels reliés entre eux et de points géométriques (déduts à partir de la position des premiers et reliés à eux par des liaisons indéformables, articulaires ou plastiques). Les points géométriques sont passifs. Ils se comportent comme des points de masse nulle.

En 1989, (Jimenez, 1993) généralise ce modèle en introduisant la modélisation par agglomérat. Les éléments constituants, appelés ici les billes, deviennent des points matériels entourés d'une zone impénétrable. Un objet est modélisé par couche de billes de plus en plus fine de l'intérieur vers l'extérieur. Les différentes couches sont :

- le noyau (grosses billes peu nombreuses simulant un comportant plutôt rigide),
- le derme (billes attirées par le noyau mais se repoussant les unes des autres, elles se répartissent uniformément autour du noyau modélisant la matière très déformable),
- l'épiderme (billes les plus petites assurant la cohérence de l'ensemble par des interactions de tension de surface, elles seules sont utilisées pour l'interaction entre objets).

L'implémentation de ce modèle est très simple (et a d'ailleurs été programmée sur une machine vectorielle) puisque les éléments individuels sont minimaux et évoluent grâce à la discrétisation des lois de la dynamique. On peut faire le parallèle avec (Terzopoulos et Witkin, 1988) car ici aussi l'objet est dissocié en une partie plus rigide et une autre plus soumise aux déformations élastiques.

Le problème posé par ce genre de modèle vient du nombre de particules nécessaires pour modéliser une forme géométrique exacte.

Robot Φ (Joukhadar, 1997)

(Joukhadar, 1997) a comme objectif la modélisation d'un robot et de son environnement afin d'améliorer le contrôle de la commande du robot. Les éléments individuels sont des points matériels reliés par des ressorts (ressorts linéaires entre deux points de masse, ressorts angulaires de torsion entre trois points de masse et ressorts angulaires de jointures entre quatre points de masse). Les points de masse discrétisent de manière volumique les objets, le découpage tétraédrique permet de respecter au mieux la matrice d'inertie des objets (la simulation est plus juste). Des paramètres permettent de contrôler la discrétisation des lois de la dynamique (identique à la méthode de Newton-Cotes, mais les coefficients 0,5 entre les vitesses sont contrôlés par l'utilisateur qui peut les faire varier). De plus on utilise un terme supplémentaire de la série de Taylor.

Autres modèles

Il existe de nombreux autres modèles entrant dans cette catégorie. On peut citer le modèle OSEA (Gascuel, 1990) qui est composé d'un noyau rigide et d'une membrane reliée au noyau à l'aide de ressorts à axes fixes. La déformation est propagée d'un ressort à ses voisins grâce à une combinaison linéaire de deux propriétés extrêmes : déformation à surface constante et

déformation à volume constant. Les éléments de la surface ne sont pas reliés entre eux par des ressorts. Les collisions sont traitées plus aisément. On intègre les lois de la dynamique.

b) Système de particules

Dans un système de particules, chaque élément (appelé ici « particule ») ne connaît pas ses voisins. Les systèmes de particules furent développés initialement pour modéliser de l'herbe ou des explosions (Reeves, 1983). (Szeliski et Tonnesen, 1992) étendent le système en orientant les particules et en ajoutant des interactions entre deux particules lorsque celles-ci sont assez proches l'une de l'autre (pour plus de détail voir (Szeliski et Tonnesen, 1991)). Les interactions suivent une loi d'attraction-répulsion de type Lennard-Jones.

(Lombardo, 1996) a développé une loi du même type mais plus robuste en terme d'oscillation. De plus il ajoute d'autres lois d'interaction permettant aux particules de mémoriser la forme locale de l'objet qu'elles représentent. Le comportement élastique est donc modélisé. En utilisant ce formalisme il a aussi développé une méthode semi-automatique de discrétisation d'objet à partir d'un champ de potentiel.

c) Surfaces implicites

On trouvera dans (Gascuel, 1993), (Desbrun et Gascuel, 1994) et (Gascuel et Desbrun, 1997) les développements essentiels des surfaces implicites. L'objet n'est plus défini, comme dans les deux cas précédents, par un ensemble d'éléments à sa surface ou dans son volume, mais comme l'union de composants solides (définissant le squelette) et d'une partie déformable définie implicitement par une équation attachée aux composants du squelette. Chaque squelette est muni d'une fonction de potentiel qui décroît plus on s'éloigne du squelette. La surface est définie par la valeur 1 de la fonction de potentiel (on peut considérer ce modèle comme une généralisation continue du modèle OSEA). L'élasticité des objets s'exprime grâce au gradient de la fonction de potentiel (on utilise le potentiel de déformation normal à la surface).

Un certain nombre de problèmes ont été résolus :

- Collisions entre deux surfaces implicites : une zone de contact entre deux objets est définie par l'intersection de leurs deux parties implicites ce qui rend le traitement beaucoup plus facile à exprimer (la surface de contact est définie de manière mathématiquement exacte). Le traitement d'une collision entre deux objets nécessite de ne discrétiser qu'un seul des deux objets.
- Changements de topologie : la discrétisation d'une surface implicite peut être optimisée grâce à une technique d'échantillonnage adaptative par graines (Desbrun et Gascuel, 1994).
- Mélange indésirable de surfaces implicites représentant deux parties distinctes d'un même objet (exemple : le bras et la jambe d'un personnage). On emploie ici un graphe de mélange indiquant à chaque squelette la liste des squelettes pouvant se mélanger avec lui.
- Déformation de fluides à volume quasi constant : chaque squelette contrôle la variation locale de son volume et modifie sa fonction de potentiel en conséquence (Gascuel et Desbrun, 1997).

Les surfaces implicites sont surtout adaptées pour la modélisation de fluide (la séparation en plusieurs parties est beaucoup plus simple à traiter). La structure en couches (une partie rigide, de type noyau ou squelette et une partie déformable rattachée) permet de construire aisément

des habillages d'objets. La discrétisation adaptative autorise une gestion simple des changements de topologie.

2.1.3 Conclusion

Nous avons examiné différentes méthodes publiées dans la littérature sous l'angle de quatre critères : principe de la loi d'évolution, stockage et manipulation de la forme, opérateur de différence entre l'objet au repos et l'objet déformé. Nous résumons dans le Tableau 1 le classement que nous avons fait plus haut.

L'avantage des modèles discrétisés est qu'ils sont connus et utilisés depuis longtemps en mécanique et en ingénierie. L'inconvénient provient soit du temps de calcul prohibitif (double discrétisation spatiale et temporelle dans le cas des différences finies, calcul de la matrice de raideur dans le cas des éléments finis), soit, essentiellement dans le cas des éléments finis, du calcul séparé du mouvement et de la déformation (on adopte alors l'hypothèse de systèmes quasi statiques).

L'avantage des modèles d'assemblage est l'utilisation d'éléments individuels. Elle permet à des degrés divers de prendre en compte des comportements non hétérogène des objets, des changements de topologies, et de simplifier les traitements tels que les collisions. D'autres part la mise en œuvre est assez simple. L'inconvénient est le nombre d'éléments à gérer pour une simulation donnée.

Tous ces modèles sont fondamentalement confrontés aux mêmes problèmes dus à la gestion discrète des objets, notamment :

- la modélisation de la forme étant approximée, il faut trouver le meilleur compromis entre nombre d'éléments et exactitude de la représentation.
- L'objet modélisé étant approximé, ses propriétés le sont aussi. Il faut donc trouver des valeurs de paramètres permettant d'obtenir le même comportement que l'objet réel.

Remarquons que le comportement de l'objet est directement lié à sa discrétisation (et ce quelle que soit l'approche utilisée). Dans notre cas, nous utiliserons un réseau de masses ponctuelles et nous exprimerons la forme avec des paramètres géométriques.

Nous verrons au paragraphe 2.2 ce que proposent les méthodes globales.

Modèles	Principe de la loi d'évolution	Manipulation de la forme	Forme de repos	Opérateur de différence
Différences finies : (Terzopoulos et al., 1987) (Terzopoulos et Fleischer, 1988) (Terzopoulos et Witkin, 1988)	Formulation de Lagrange	Discretisation régulière (différences finies)	Tenseur métrique et de courbure	Dérivée de l'énergie de déformation
+ objet de référence				
Éléments finis : (Gourret et al., 1989) (Bainville, 1996)	Travaux virtuels + Minimisation d'énergie	Discretisation en éléments puis en nœuds (éléments finis)	Position et matrice de raideur	Vecteur déplacement
Réseaux explicites : (Miller, 1988) (Jimenez, 1993) (Joukhadar, 1997)	Intégration de la loi fondamentale de la dynamique (LFD)	Masses-ressorts	Longueur des ressorts	Longueur des ressorts
		Aggrégats Masses-ressorts	Liaisons de type ressorts linéaires	Ressorts individuels
Système de particules : (Lombardo, 1996)	Intégration des forces d'attractions	Particules + surfaces implicites	Lois de type interactions (+ typage)	Forces d'attraction (fonction de la distance)
Surfaces implicites : (Gascuel et Desbrun, 1997)	Intégration de la LFD et des lois de déformation	Squelette et fonction de potentiel	Fonction de potentiel initial	Différence du gradient due au potentiel

Tableau 1. Les approches nodales.

Classement des méthodes selon les critères. Le principe de la loi d'évolution exprime la formulation choisie pour la loi d'évolution. Le critère de manipulation de la forme exprime comment est manipulé l'objet (la plupart du temps sa surface). Le critère de forme au repos décrit comment est stockée la forme de l'objet au repos (critère souvent utilisé aussi dans l'opérateur de différence). L'opérateur de différence explique comment est calculée la différence entre l'objet au repos et l'objet déformé.

2.2 Approches globales

Pour réduire les coûts calculatoires des approches nodales, des approches globales ont été développées. Deux méthodes sont généralement utilisées : la première manipule seulement les déformations globales, la deuxième ne prend en compte que certains modes vibratoires.

2.2.1 Déformations globales

Il existe d'autres modèles de déformations globales plus générales et ne s'appliquant pas forcément aux modèles physiques (voir notamment (Sederberg et Parry, 1986) et (Watt et Watt, 1992) pour les déformations de formes libres, FFD). Elles sont utilisées essentiellement dans le cadre de modéleur d'objets et plus récemment dans le cadre d'animation d'objets déformables.

On peut apparenter la solution de (Witkin et Welch, 1990) à ces méthodes de déformation globales. Les auteurs restreignent l'ensemble des déformations à des transformations matricielles particulières. Les déformations sont globales : les éléments composant les objets sont plongés dans l'espace où s'appliquent les déformations. Dans cet espace, on définit une énergie de déformation permettant de simuler des comportements élastiques ou de compressibilité. La formulation de Lagrange est utilisée, elle permet notamment de résoudre des contraintes par la méthode des forces de contraintes (voir paragraphe 3.2). Dans cet article, l'accent n'est pas mis sur les performances de simulation d'une telle modélisation, mais plutôt sur la facilité d'obtenir des déformations rattachées à la physique (le comportement élastique n'y est pas vraiment présenté) et d'inclure ce formalisme dans des résolutions classiquement utilisées pour la simulation physique (notamment la résolution de contraintes et des zones impénétrables (Barraf et Witkin, 1992)).

Les auteurs utilisent des déformations affines (translation, rotation, étirement, torsion) et une formulation d'énergie potentielle de déformations pour produire des forces de déformations. La méthode de (Terzopoulos et Metaxas, 1991), basée sur les mêmes principes, utilise des superquadriques qu'ils déforment globalement, mais aussi localement, pour la segmentation ou la reconnaissance de forme.

Le modèle DOGME, voir (Borrel et Bechmann, 1991) et (Bechmann et Dubreuil, 1993), est un modèle de déformation de forme libre pouvant s'appliquer à des polyèdres. Une déformation est définie par des contraintes de déplacements. Chaque contrainte, définie par l'utilisateur, force un point de l'espace tridimensionnel à un déplacement. Par l'intermédiaire d'une fonction d'extrusion choisie dans une bibliothèque (par exemple le produit tensoriel de B-splines dont les nœuds pavent l'espace euclidien) une matrice de projection est calculée. Cette matrice de projection déforme l'espace, l'objet géométrique considéré est ainsi déformé. Le calcul de la matrice de projection fait apparaître un terme d'optimisation qui permet de définir des conditions de déformation comme l'attraction ou la répulsion d'un point de l'espace (Borrel et Bechmann, 1991), ou la préservation du volume (Aubert et Bechmann, 1995) et (Aubert et Bechmann, 1997).

2.2.2 Déformations modales

Dans (Pentland et Williams, 1989), les auteurs utilisent la méthode des éléments finis et la formulation lagrangienne des lois d'évolutions. En posant $\mathbf{C} = k\mathbf{M}$, on peut remarquer que les trois matrices \mathbf{M} , \mathbf{C} et \mathbf{K} (normales définies positives symétriques) admettent une base

commune de diagonalisation. On se sert de cette diagonalisation pour obtenir un ensemble de $3n$ équations indépendantes (si n est le nombre de points ou de nœuds d'éléments finis utilisés). Chaque équation indépendante régit un mode vibratoire particulier. Pour calculer les déformations des objets en rejetant les déformations de hautes fréquences (modifiant peu la forme globale de l'objet), on ne prend en compte qu'une partie des équations que l'on superpose linéairement. Pour connaître quelles équations choisir, on examine les valeurs propres de la base de diagonalisation correspondantes aux équations : la valeur propre est proportionnelle à la fréquence.

Les avantages essentiels de telles approches sont :

- la diminution du nombre de degrés de liberté du système (le contrôle de la déformation est donc plus aisé),
- la globalisation des calculs : les problèmes dus à la discrétisation (notamment divergence des modèles) sont moins prédominants.

Les inconvénients sont que les déformations d'objets hétérogènes ne sont pas possibles, et que les déformations locales sont mal simulées. De plus, ces modèles ne sont pas adaptés à la modélisation précise d'un phénomène : il faudrait caractériser les déformations du phénomène et trouver leur expression en terme de déformation globale (ce qui n'est pas toujours possible). Pour cette raison nous choisirons de suivre l'approche nodale, le reste de ce chapitre s'applique principalement à cette dernière.

3. Contraindre les modèles

Du fait de leur importante flexibilité et des possibilités nombreuses de déformation qu'ils comportent, les modèles déformables élastiques ont rapidement nécessité le développement d'outils et d'algorithmes de contrôle. Nous considérerons que contrôler un modèle revient à restreindre son domaine de déformation, donc à ajouter des contraintes au modèle. Nous verrons dans un premier temps les contraintes nécessaires à un modèle, puis les différentes méthodes de résolution de ces contraintes proposées dans la littérature.

3.1 Contraindre

Par contraindre on entend plutôt maîtriser un certain aspect de l'évolution d'un objet. C'est-à-dire réduire l'espace de liberté de l'objet afin de contrôler son évolution. On s'aperçoit en effet qu'outre les paramètres de discrétisation (spatial ou temporel) et les paramètres de déformation (coefficients appliqués aux forces), il faut, pour modéliser un phénomène ou un système précis, réduire le nombre de déformations possibles.

Il existe deux types de contraintes (on appelle ici élément un nœud de discrétisation) :

- **Contraintes locales** : la contrainte est appliquée sur un ensemble restreint d'éléments. C'est le type de contrainte le plus courant (et souvent le plus nécessaire). On peut citer les exemples les plus courants : un élément est fixé dans l'espace, deux éléments doivent toujours être confondus, aucun élément ne doit pénétrer dans une région particulière de l'espace, un élément doit suivre un certain chemin défini par l'utilisateur, l'élément manipulé par l'utilisateur doit suivre les mouvements captés par un périphérique

interactif, un élément doit toujours rester dans une région de l'espace, un élément doit se déplacer à une vitesse \mathbf{v}_i de $t = t_0$ à $t = t_1$.

- **Contraintes globales :** dans ce cas un ensemble important d'éléments de l'objet, voire tous les éléments, doivent vérifier une seule contrainte (c'est-à-dire une contrainte définie globalement sur cet ensemble). Le cas le plus intéressant est la contrainte de déformation à volume constant. Outre la modélisation d'objets incompressibles, elle permet dans la plupart des cas, d'assurer directement la conservation de la masse d'un objet (dans le cas d'un objet homogène et de densité constante). Cela peut être utile dans les modélisateurs de Conception Assistée par Ordinateur. On peut citer aussi la contrainte de déformation à surface constante. On peut englober dans ce type de contrainte celles qui s'appliquent de manière temporelle. Par exemple on veut qu'un objet suive le chemin « le plus court » entre son point de départ et son point d'arrivée (on peut se servir de ce type de contrainte dans une phase d'apprentissage ou de caractérisation des paramètres en fonction d'un comportement ciblé).

On peut parler de formulation faible ou forte pour une contrainte. Dans le cas de formulation faible, la contrainte sera satisfaite « au mieux » (une certaine erreur est tolérée). Dans le cas de la formulation forte, la contrainte sera toujours satisfaite exactement (à la précision de la machine près).

En ce qui concerne les approches d'assemblage d'éléments, le modèle manipule directement les éléments des objets. Il apparaît donc que ce type d'approches est plus aisée à contraindre : on peut appliquer une contrainte dès que l'on sait la traduire en terme de position, de vitesse ou de force sur les éléments. Nous nous sommes donc porté naturellement vers ces approches.

De nombreuses méthodes ont été développées pour contraindre les déplacements d'objets rigides (notamment pour les solides articulés), voir paragraphe 3.2. On s'intéresse ici plutôt aux méthodes appliquées ou applicables à des objets déformables.

Notons que les contraintes peuvent aussi servir à créer des objets par assemblage. En dictant un certain nombre de lois de comportement (Lombardo, 1996) laisse les particules se distribuer à la surface des objets tridimensionnels définis par des potentiels. Une contrainte point-à-point est aussi employée dans (Gascuel et Gascuel, 1994) pour créer des objets rigides à partir de pièces rigides plus simples, de même que dans (Barzel et Barr, 1988).

Dans le paragraphe suivant, nous examinons les différents moyens d'ajouter des contraintes.

3.2 Les différentes méthodes de résolution de contraintes

On peut classer les méthodes de résolution en trois catégories : les méthodes d'optimisation cherchant à optimiser une erreur sur la contrainte par l'intermédiaire de méthodes mathématiques classiques, les méthodes cherchant les forces exercées par les contraintes et enfin les méthodes cherchant les déplacements dus aux contraintes. Nous examinons ces trois classes de méthode dans les paragraphes suivant (paragraphe 3.2.1, 3.2.2 et 3.2.3).

Une étude plus détaillée de la littérature concernant la résolution de la contrainte d'incompressibilité est fournie au chapitre 4 du présent manuscrit. On y étudie les travaux de

(Platt et Barr, 1988), (Rappoport et al, 1996), (Aubert et Bechmann, 1997) et (Gascuel et Desbrun, 1997).

3.2.1 Méthodes d'optimisation

(Platt et Barr, 1988) font la remarque suivante : un modèle élastique est un système qui minimise une fonction. Cette fonction est la fonction énergétique du modèle. C'est pourquoi, il semble logique d'explorer les méthodes mathématiques de minimisation de fonction sous contraintes (par exemple les méthodes de descentes de gradient).

a) Pénalités

La méthode des pénalités est une des plus connues. Elle est utilisée notamment chez (Terzopoulos et al., 1987) et ses dérivées, (Pentland et Williams, 1989) ou encore (Dumont et al, 1989). Il s'agit en fait d'ajouter un terme d'énergie quadratique pénalisant les déplacements qui éloignent de la contrainte. Cette méthode a l'avantage d'être très simple à implémenter. Par contre, outre le fait que la contrainte n'est pas vérifiée exactement, en augmentant le poids de la pénalité (par exemple pour diminuer la déviation de la contrainte) on change le comportement du système (il devient plus rigide, le pas de temps doit être diminué).

b) Multiplicateur de Lagrange et méthode des lagrangiens augmentés (Platt et Barr, 1988)

On exprime la minimisation sous contrainte en cherchant un point critique de l'équation énergétique définie par la fonction à minimiser (énergie totale du système) additionnée de l'équation de contrainte pondérée par un coefficient (le multiplicateur de Lagrange). En fait, on introduit de nouvelles équations différentielles permettant de prendre en compte la contrainte. Pour résoudre ce problème on ne peut pas utiliser une méthode classique de descente de gradient : ici on se dirige dans la direction inverse, en cherchant un minima. La méthode des lagrangiens augmentés est la combinaison de la méthode des multiplicateurs de Lagrange et de la méthode de pénalités. Elle permet de résoudre des contraintes d'inégalités. Ce type de résolution peut prendre en compte de nombreuses contraintes. Dans (Platt et Barr, 1988) on calcule par cette méthode des déformations quasi incompressibles ou des comportements plastiques.

Les inconvénients sont d'une part que la contrainte n'est pas exactement vérifiée et d'autre part que des équations supplémentaires sont ajoutées au système.

Dans le chapitre 4, nous comparerons cette méthode, dans le cadre d'un problème simple de minimisation sous contrainte avec la méthode proposée par (Fua et Bechbühler, 1997) et avec la méthode développée dans cette thèse.

3.2.2 Méthodes de recherche des forces exercées par les contraintes

Les deux premières méthodes que l'on présente ici s'appliquent à des solides articulés indéformables. Pourtant elles peuvent être transposées à des objets déformables. Il s'agit ici de calculer les forces dues aux contraintes (soit en intégrant ce calcul directement dans les équations d'évolution, soit en découplant ces forces des autres forces). Ces forces sont celles qui, issues des contraintes, influencent l'évolution des objets pour qu'ils vérifient ces contraintes.

a) Dynamique inverse (Barzel et Barr, 1988)

En dynamique inverse, on calcule à chaque instant à partir de l'état du système (positions, vitesses et forces) la force qui va soit amener à vérifier la ou les contraintes, soit qui va maintenir la contrainte. Pour cela on utilise un opérateur de différence entre l'état actuel et l'état vérifiant les contraintes : c'est la mesure de la déviation de la contrainte, prenant en compte position, vitesse et accélération. À partir de cet opérateur de différence, on obtient une équation différentielle dont l'inconnue est la force de contrainte à appliquer pour minimiser cette déviation.

b) Approche itérative (van Overveld, 1991)

Ici on dissocie les forces de contraintes des autres forces (non dues aux contraintes). Ainsi, il est inutile de résoudre les systèmes d'équations du mouvement et des contraintes (comme dans la méthode de (Barzel et Barr, 1988)) : il suffit, par une méthode itérative de calculer l'action des forces dues aux contraintes (sur l'accélération, la vitesse et la position) jusqu'à obtenir la convergence (vérification des contraintes). Cette approche est appliquée pour des contraintes géométriques sur un ensemble de masse ponctuelles (une barre est modélisée par deux points se maintenant constamment à la même distance l'un de l'autre). Elle semble moins générale que la précédente mais beaucoup plus simple et beaucoup plus rapide à calculer.

c) Réaction dynamique (Platt et Barr, 1988)

C'est une méthode également très utilisée. Dans la littérature on ne l'applique que pour résoudre des contraintes locales comme faire suivre un chemin ou réagir à la pénétration. Dès qu'une contrainte n'est pas vérifiée, une méthode de projection est utilisée pour calculer la ou les forces nécessaires à la vérification de la contrainte. Par exemple si un point doit rester coller à un polygone, dès que d'autres forces le déplacent en dehors, on calcule la force qui ramène ce point sur le polygone selon la projection. Cette technique est plus rapide que la méthode des pénalités mais interdit tout calcul de la réaction à la force déclenchant la contrainte.

3.2.3 Méthodes de recherche des déplacements dus aux contraintes

Dans ce cas, on ne calcule que l'effet direct des contraintes sur les positions. Cette méthode s'applique surtout aux approches d'assemblage d'éléments. Dans la littérature cette méthode est appliquée au solides articulés (Gascuel et Gascuel, 1994) sous forme d'un calcul itératif ou pour contraindre des déformations de tissus (Provot, 1995) par l'utilisation directe du déplacement (en une seule passe).

a) Calcul itératif (Gascuel et Gascuel, 1994)

On part du principe de découplage des forces dues aux contraintes et des autres forces. Cependant on élimine la dynamique de la résolution : on ne va pas calculer les forces dues aux contraintes. Les contraintes seront vérifiées de manière géométrique en effectuant itérativement des petits déplacements des nœuds des objets. Dans cet article, la méthode est utilisée pour des solides articulés. Les données physiques des objets sont prises en compte lors de la résolution des contraintes ce qui permet d'obtenir un meilleur réalisme. C'est un

moyen rapide de résoudre des contraintes sur des structures articulées. Elle correspond à l'utilisation de ressorts pour modéliser les contraintes.

b) Utilisation directe de déplacement (en une seule passe)

Généralement employée pour des contraintes locales (comme substitut au calcul de la réaction dynamique), cette méthode revient à calculer la position dans l'espace vérifiant la contrainte pour l'ensemble des points contraints. Cela n'est pas toujours possible (par exemple dans le cas de rigides articulés). Dans (Provot, 1995) on l'utilise pour contraindre l'élongation des ressorts : dès qu'un seuil d'élongation maximum est dépassé, les éléments du maillage incriminés sont déplacés afin de vérifier la valeur d'élongation.

La méthode directe a notamment été développée au cours de cette thèse pour vérifier des contraintes globales comme l'incompressibilité (Promayon et al, 1996). Nous verrons dans la partie suivante de ce document un exemple précis de résolution directe. Ici aussi, les forces dues aux contraintes sont découplés des autres forces, mais au lieu de calculer itérativement des petits déplacements, la contrainte est directement résolue à l'aide du gradient. On peut la considérer comme une méthode de réaction appliquée à une contrainte globale utilisant directement les déplacement liés à la vérification des contraintes. L'avantage est la rapidité. Une étude mathématique a permis de valider physiquement cette méthode dans le cas de la contrainte d'incompressibilité.

4. La problématique actuelle

On a vu dans les paragraphes précédents la base des outils de simulation physique (lois fondamentales, modèles déformables et contraintes). Ce paragraphe passe en revue un certain nombre de problèmes qui sont actuellement l'objet de recherches. L'étude bibliographique faite aux paragraphes précédents ne serait pas complète si nous ne présentions pas cette problématique. L'étude menée au cours de cette thèse n'a pu se faire sans prendre cette problématique en compte et y apporter des solutions.

Ce paragraphe décrit brièvement l'état de l'art dans les domaines suivant :

- La simulation d'objets complexes hétérogènes (paragraphe 4.1),
- La reconstruction d'objet à partir de données réelles comme base géométrique des objets simulés (paragraphe 4.2),
- La paramétrisation des modèles (paragraphe 4.3).

Le paragraphe 4.4 décrit d'autres problématiques à considérer dans notre étude.

4.1 Simulation d'objets complexes hétérogènes

Historiquement, c'est le développement de modèles déformables qui a d'abord intéressé la communauté scientifique (peu de travaux avaient été effectués dans le domaine de la simulation physique). Pourtant, on s'est vite rendu compte que créer une scène ou une animation demandait l'intégration de modèles déformables et de modèles de solides (articulés ou non) au sein de modèles complets.

Le moyen le plus simple de plonger un modèle déformable dans une scène comportant des objets rigides est l'utilisation de contraintes : contraintes de non-pénétration dans les solides (réaction dynamique), contraintes de pénalité (les solides sont entourés d'un potentiel

pénalisant l'énergie du modèle déformable). Cependant cette solution ne permet pas de restituer l'interaction entre un objet solide et un objet déformable. On veut par exemple modéliser un instrument chirurgical explorant des organes (on a besoin de calculer la déformation induite par l'instrument sur les objets déformables que sont les organes).

Il a donc fallu développer des techniques d'interactions. Deux cas se présentent :

- Approche transversale : une seule technique permet de modéliser les interactions entre les objets déformables et les objets rigides (on applique alors cette technique aux deux types de modèle),
- Approche superviseur : on utilise plusieurs modèles que l'on contrôle par un superviseur (qui va « traduire » et faire la passerelle entre deux modèles) au travers d'une plate-forme.

4.1.1 Approche transversale

À l'ACROE (Grenoble), on utilise la notion d'agrégat (présenté au paragraphe 2.1) pour décrire des objets plutôt rigides (comme le sol). Tous les objets étant modélisés avec le même formalisme, les interactions sont automatiquement produites. Le problème d'un tel modèle est le nombre d'éléments de discrétisation et de relations entre ces éléments à mettre en place pour modéliser une scène.

(Bainville, 1996) a développé un système à deux dimensions utilisant les éléments finis pour calculer les déformations d'objets élastiques. Les objets solides y sont représentés sous forme polygonale. L'interaction entre objets solides et déformables est calculée automatiquement par un seul formalisme (utilisation de curseurs qui discrétisent la surface des objets pour la détection et le traitement des collisions).

De manière générale, de nombreux projets font état de l'interaction entre un modèle déformable spécifique (discrétisé en masses ponctuelles) et un modèle de solides (discrétisé en point de masses en utilisant la dynamique des solides, voir annexe 1).

On distingue les approches utilisant les deux modèles en tant que tels (comme (Joukhadar, 1997)), des modèles hybrides qui modélisent un même objet en utilisant deux composants (composant rigide pour le mouvement et composant déformable). Dans cette dernière catégorie on peut classer tous les modèles s'appuyant sur une couche de peau (modèle déformable) attachée (souvent par de simples ressorts) sur une structure osseuse (modèle rigide) : voir (Gascuel et Desbrun, 1997), (Terzopoulos et Fleischer, 1988), (Terzopoulos et Witkin, 1988), (Chadwick et al., 1989), (Meseure, 1997).

4.1.2 Plate-forme

Dans le projet SIAMES de l'IRISA à Rennes, un modèle unifié, appelé DREAM (Deformable and Rigid objects Efficient for Animation Modelling), est développé : voir (Cozot et al., 1995) et (Dumont et al., 1989). C'est un exemple de travail où on ne cherche pas à trouver un modèle universel, mais où une plate-forme permettant l'intégration de différents types de modèles est proposée. Un formalisme abstrait définit les modèles existants (ici modèle des solides, éléments finis et réseaux masses-ressorts) autour des notions d'entité et de relation. La formulation lagrangienne et le principe des travaux virtuels sont employés par des

agents dédiés à un type de modèle pour calculer l'évolution des objets. Cela permet de choisir le meilleur modèle en fonction du type d'objet que l'on veut modéliser.

4.2 Base géométrique des objets simulés

Lorsqu'on modélise un objet, et quelle que soit l'approche que l'on choisit, il faut au préalable une base géométrique de l'objet. Dans le cadre général on peut utiliser un logiciel spécialisé (modeleur CAO) qui utilise des techniques particulières pour créer des objets (par assemblage comme les CSG, déformation par l'intermédiaire de FFD ou déformations globales affines). Par contre, dans le cadre d'un simulateur, on a besoin du modèle géométrique précis d'un objet particulier. Dans la problématique GMCAO on a besoin du modèle d'un organe particulier. Pour obtenir la géométrie de cet objet on utilise des données issues d'atlas anatomiques (mais très peu existent aujourd'hui en 3D) ou d'images médicales (acquisition sur un patient ou un sujet précis). Se pose alors le problème de la segmentation d'images (Ayache et al., 1996).

Au sein du laboratoire TIMC de nombreuses techniques ont été développées qui utilisent des contours déformables (de type snake) attirés par un potentiel lui-même obtenu à partir des données de l'image. On peut citer notamment la technique des snake-splines de F. Leitner (Leitner, 1993) et celle des δ -snake (Lachaud et Bainville, 1994). Un des principaux problèmes est la reconstitution tridimensionnelle à partir des séquences d'images des appareils d'imagerie ou à partir de contours pris sous des angles différents (si on utilise un balayage laser, voir (Bittar et al, 1993)).

Le choix d'un modèle de déformation doit être influencé par la façon dont les données géométriques utilisées en entrée de ce modèle peuvent être obtenues. Dans la majorité des cas le contour facettisé peut être obtenu en sortie des algorithmes de segmentation et de reconstruction. De manière générale, une approche par assemblage facilitera le lien entre les données issues d'un modeleur ou d'algorithmes de segmentation car la discrétisation est déjà présente.

4.3 Paramétrisation des modèles

Une fois la forme géométrique de l'objet obtenu, il faut encore utiliser le modèle de déformation. Pour cela il faut trouver les paramètres correspondants aux comportements que l'on veut modéliser. Cette phase est assez délicate puisque peu de méthodes génériques existent. On peut les classer en deux groupes : les méthodes de minimisation d'erreur (méthode des moindres carrés, ou même utilisation d'un réseau de neurones) et les méthodes probabilistes (notamment algorithmes génétiques). Nous verrons cela plus en détail dans la troisième partie de ce manuscrit.

4.4 Autres problèmes

Ce paragraphe présente d'autres problématiques. Elles sont examinées ici par ordre décroissant de nos préoccupations. On examinera :

- La modélisation des muscles,
- La stabilité des modèles,
- Le degré d'interactivité des modèles,

- La simulation et l'interaction de modèles d'humains dans des mondes virtuels.
- La gestion des contacts, collision et des changements topologiques,

4.4.1 Modélisation des muscles

Un état de l'art concis couvrant les principales modélisation des muscles par des modèles générateurs sera présenté au chapitre suivant lorsque nous aborderons la construction de la propriété musculaire. On verra notamment que deux approches sont généralement employées :

- les approches simulant la déformation : (Chadwick et al., 1989) (Volino et al, 1996) et (Wilhelms, 1997),
- les approches simulant le fonctionnement : (Waters, 1987), (Lee et al., 1995), (Beylot et al, 1996), (Maurel et al, 1996), (Miller, 1988), (Chen, 1992) et (Lombardo, 1996).

4.4.2 Stabilité (méthode d'intégration vs modèle informatique)

La stabilité d'un modèle déformable peut se mesurer par l'intermédiaire :

- De son champ de modélisation : plus le modèle est stable, plus il va pouvoir modéliser des comportements extrêmes.
- Du pas de temps maximal qu'il supporte : plus le modèle est stable, plus il pourra supporter des pas de temps élevés. Ceci est important notamment dans le cadre de simulation en temps réel ou en temps interactif.

L'utilisation d'un ordinateur nous oblige à faire des calculs discrets. C'est pourquoi la stabilité numérique doit guider notre choix du modèle. En effet, pour éviter la divergence d'un modèle on peut :

- choisir une méthode d'intégration plus précise (voir paragraphe 1.4), ou
- choisir un modèle moins sujet aux oscillations.

Ce dernier point est assez important, c'est pourquoi, dans notre approche nous avons essayé de minimiser les éléments à caractère oscillant (de type ressort) ; c'est ce que nous verrons dans la deuxième partie de ce manuscrit.

Les modèles générateurs font souvent une utilisation « abusive » de la viscosité. Les forces de viscosité, opposées à la vitesse des objets ou de leurs éléments, permettent de ralentir les déplacements, induisant souvent une plus grande stabilité. C'est pourquoi les forces de viscosité sont souvent amplifiées au-delà de la simple modélisation du phénomène physiques qu'elles représentent.

4.4.3 Interactivité (problématique GMCAO)

Ce problème rejoint le paragraphe 4.1. En effet, dans la plupart des cas les modèles ont été construits pour créer une animation d'images de synthèse ou pour simuler un phénomène réel précis. Dans les deux cas, l'interaction avec l'utilisateur (qu'il soit spécialiste médical ou spécialiste en animation) est souhaitable.

L'interaction avec l'utilisateur passe toujours au travers des sens de ce dernier : tout d'abord visuel, puis tactile ou sonore. Ceci implique donc un autre périphérique (par exemple acquisition de mouvement en médecine et/ou en animation de film de synthèse, ou encore surveillance médicale).

En ce qui concerne l'implémentation, l'interactivité avec l'utilisateur nécessite donc la mise en place d'une interface entre le périphérique et le modèle (voir (Bainville, 1996) et (Simon, 1996)). Cela implique que le modèle doit être capable de réagir en fonction de l'état courant et des apports du manipulateur. Cela nécessite souvent une grosse puissance de calcul et est plus facile à obtenir si on manipule directement les nœuds de la discrétisation de manière directe. Cet aspect doit lui aussi guider la construction du modèle.

4.4.4 Simulation de l'humain dans des mondes virtuels

La première simulation d'acteurs virtuels remonte au film de synthèse *Rendez-vous à Montréal* (Magnenat-Thalmann et Thalmann, 1987) où l'on voit la réunion posthume de deux acteurs célèbres (Marilyn Monroe et Humphrey Bogart) dans un environnement de synthèse. Bien que réaliste du point de vue visuel, les acteurs sont contrôlés de manière empirique par des techniques évoluées utilisant des positions clés. Par la suite, apparaissent des simulations de la marche utilisant toujours les mêmes approches (Bruderlin et Calvert, 1989), ou développant des techniques permettant un meilleur contrôle et une individualisation de l'animation (Boulic et al, 1990). Parallèlement, des méthodes plus globales permettent de construire des humains virtuels agissant dans leur propre monde (Magnenat-Thalmann et Thalmann, 1991). Aujourd'hui les techniques se spécialisent (Hodgins et al, 1995) ou deviennent interactives (Bruderlin et Calvert, 1993). La problématique actuelle est située autour de modèles de simulation comportementale qui permettrait la simulation d'un monde où les acteurs virtuels autonomes interagiraient entre eux ou avec leur environnement par l'intermédiaire de leurs propres capteurs.

En quelques années, on est passé d'approches géométriques bas niveau à des approches comportementales (par l'intermédiaire de modèles générateurs) dans lesquels il est possible d'individualiser le comportement de personnages virtuels. Le modèle présenté dans cette thèse peut, dans ce sens, être directement accroché à cette problématique.

4.4.5 Gestion des contacts, collisions, et changements topologiques

Dans ce document il ne sera pas fait état des problèmes d'interaction liés aux accidents entre les objets eux-mêmes (frottements et détection et traitement des collisions). Pour cela on se référera à des articles sur le contact (Baraff, 1994) ou (Faure, 1996), les collisions (Moore et Wilhelms, 1988) ou (Joukhadar, 1997) pour une bibliographie complète, les déchirures et fractures (Terzopoulos et Fleischer, 1988) ou (Meseure, 1997) pour une bibliographie complète.

5. Conclusion

Au chapitre 1, nous avons examiné le phénomène à modéliser (les mouvements du tronc pendant la respiration) pour en dégager les structures à prendre en compte (cage thoracique, abdomen, poumons et diaphragme). Ces structures sont représentables par des régions solides, des régions élastiques, et des régions musculaires. Le tronc est alors considéré comme un objet complexe composé de plusieurs régions aux caractéristiques différentes interagissant entre elles. Ces régions, leurs interactions et leurs propriétés physiques doivent maintenant être décrites de manière plus précise.

Nous avons vu au cours de ce chapitre comment obtenir une modélisation comportementale satisfaisante. Cela passe par l'utilisation de modèles générateurs. Parmi les deux approches possibles de modèle générateur basé sur la physique nous avons montré que l'approche nodale (et plus particulièrement l'approche par assemblage d'éléments) était mieux adaptée à la description du tronc tel que nous le considérons : facilité d'implémentation, définition d'objets hétérogènes, contrôle et contrainte directe sur les nœuds, interaction entre objets rigides et objets déformables, couplage d'un périphérique plus simple, applicabilité des algorithmes de traitement des contacts, des collisions et des changements topologiques.

Cependant, nous n'avons trouvé dans cette approche aucune méthode de la littérature capable de répondre à nos besoins. Nous allons présenter dans la partie suivante le modèle que nous avons développé qui permet de manière homogène de définir des objets déformables, élastiques, actifs, incompressibles et de les lier avec des objets solides.

Chapitre 3

Construction du modèle

*Yukti yuktam upâdeyam vacanam bâlakâd-âpi/anyat tr. n
am iva tyâjyam âpy-uktam padmajanmana.¹*

Svâmiiji Prajnânpad, mantra sanscrit.

Dans ce chapitre nous établissons les bases de notre modèle générateur par la méthode d'assemblage. Comme nous l'avons vu au chapitre 1, notre objectif est de modéliser des propriétés très différentes les unes des autres : des tissus élastiques (représentant les tissus mous du corps humain) ; les tissus musculaires, et aussi les tissus rigides (représentant les os du squelette). Modéliser les mouvements du tronc au cours de la respiration peut être vu comme la modélisation d'un objet complexe comportant plusieurs régions de propriétés différentes. Notre modèle doit donc pouvoir modéliser des objets hétérogènes complexes. La philosophie que nous avons choisie d'adopter s'adapte directement aux propriétés que nous devons modéliser en utilisant une approche informatique orientée-objet.

Pour cela, on sépare un objet hétérogène en plusieurs régions de propriétés différentes. Suivant l'approche par assemblage d'éléments, ces régions seront constituées d'éléments de propriétés identiques (voir Figure 1).

Dans ce chapitre, nous étudierons les propriétés à modéliser en commençant par fixer la base géométrique et dynamique des régions composants un objet. Ensuite nous verrons comment construire des régions élastiques et leurs éléments, des éléments et des régions musculaires et enfin des régions et des éléments solides. Dans le chapitre suivant, nous décrirons des méthodes pour contraindre ces différentes régions et leurs éléments (transmissions des forces entre région, contraintes locales et globales).

¹ Ce qui est raisonnable accepte-le immédiatement, même si c'est un enfant qui le dit. Ce qui est contraire à la raison écarte-le, même si c'est un dieu qui parle.

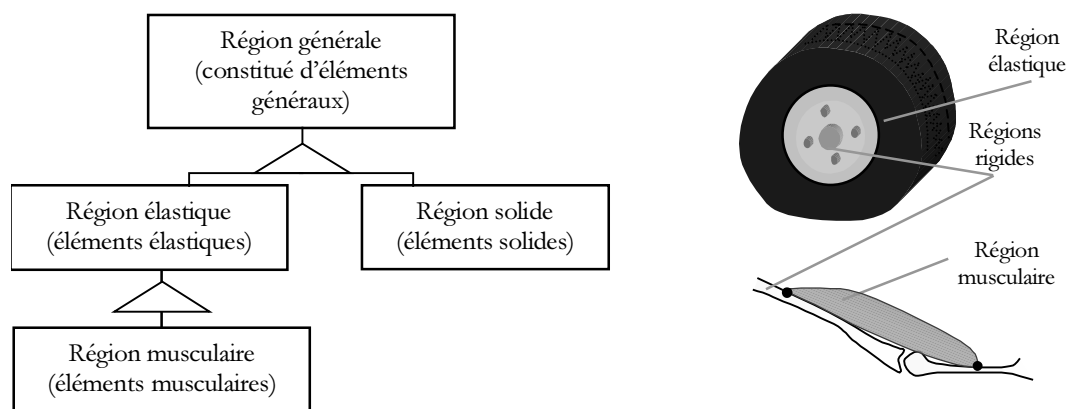


Figure 1. Structure d'un objet. Dans notre modèle, un objet est découpé en régions de propriétés distinctes. Une approche orientée-objet est utilisée pour les différentes régions (les flèches représentent les relations d'héritage). Les régions sont constituées d'éléments individuels.

1. Organisation du chapitre

En guise d'introduction, l'organisation et l'articulation de ce chapitre est décrite brièvement. Les régions générales (paragraphe 2.1) sont construites de telle façon que l'utilisation de bases de données soit possible (telles que celles issues des algorithmes de segmentation, des modeleurs 3D, des algorithmes de gestion de surfaces et des techniques d'affichage). Nous montrons ensuite comment les lois de la dynamique peuvent y être ajoutées (paragraphe 2.2). La présentation de la notion de force engendrée par un point attracteur est particulièrement importante puisqu'elle définit le principe directeur de notre modélisation des forces élastiques et musculaires.

Pour définir des régions élastiques, nous appliquons une modélisation physique à partir d'une définition originale des formes géométriques (paragraphe 3.1 et 3.2). Nous comparons ensuite notre méthode à l'approche masse/ressort classique et la situons par rapport au tableau 1 du chapitre précédent (paragraphe 3.3).

Après avoir dressé un état de l'art succinct sur les modèles de muscles de la littérature (paragraphe 4.1), on montre comment on peut appliquer un principe simple pour obtenir des propriétés complexes comme la définition de directions de contraction (paragraphe 4.2).

Nous verrons enfin que l'approche décrite au chapitre 2 et en annexe 1 pour définir les parties rigides est valable et directement utilisable dans notre modélisation (paragraphe 5). Pour chacune des régions (élastiques, musculaires et élastiques), le paragraphe final récapitule les attributs nécessaires aux éléments de ces régions.

2. Description générale

Dans ce paragraphe nous allons décrire la base géométrique des régions, ainsi que les principes de la dynamique que nous allons utiliser (description des forces et des lois d'évolution des régions).

2.1 Géométrie des objets déformables

Les régions d'un objet sont décrites par leurs surfaces composées de facettes triangulaires. Ces deux points seront argumentés respectivement au paragraphe 2.1.1 et 2.1.2 .

2.1.1 Modélisation surfacique des objets

Dans une approche par assemblage d'éléments, chaque objet est décrit de manière géométriquement discrète par un ensemble de points. Trois approches principales sont possibles comme nous l'avons vu au chapitre précédent : construction d'un réseaux, gestion d'un système de particules, ou utilisation de surfaces implicites. Nous avons opté pour la première. On choisit de plus de décrire un objet uniquement par sa surface. Les éléments seront donc disposés à la surface de l'objet (sa limite spatiale).

La construction d'un réseau surfacique, c'est-à-dire la définition de voisinages explicites entre les éléments composant la frontière géométrique des objets, est la représentation la plus courante que ce soit dans les modeleurs pour l'animation, les logiciels de CAO ou bien les algorithmes de segmentation. Cela nous permet de disposer de nombreux objets et de pouvoir les utiliser directement.

Enfin, pour rester cohérent avec notre approche, chaque élément de surface est indexé de manière unique à l'intérieur d'un objet. On n'utilise donc pas de représentation paramétrique des surfaces (plutôt utilisée dans les approches par différences finies).

Le pari est donc le suivant : définir les lois de d'évolution (comme l'élasticité, l'incompressibilité, la motricité) en n'utilisant que des éléments à la surface des objets.

2.1.2 Utilisation de facettes triangulaires

Nous avons choisi de relier ces éléments à travers un réseau de facettes triangulaires. Cela comporte plusieurs avantages :

- La liste de facettes triangulaires est le format le plus répandu pour décrire une surface tridimensionnelle (toutes les sources d'objet 3D citées plus haut peuvent fournir un réseau de facettes triangulaires).
- Le triangle est l'élément de surface le plus simple, son utilisation dans les calculs permet de nombreuses simplifications (comme nous le verrons au chapitre suivant pour le calcul du volume).
- Les facettes triangulaires sont des primitives graphiques présentes dans toutes les bibliothèques de représentation d'objets 3D (comme PEX ou OpenGL). Généralement câblés, les algorithmes d'affichage de ces primitives sont très performants.

Chaque élément gère la liste de ses voisins, avec lesquels il forme des facettes triangulaires. Cette liste de voisin est invariante dans notre cas, car nous n'avons pas modélisé les changements de topologie ou les fractures.

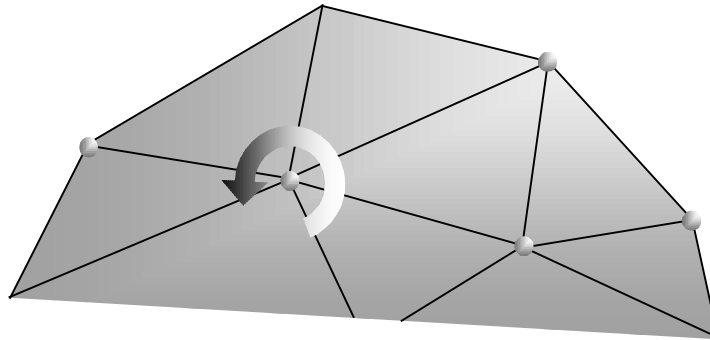


Figure 2. Représentation des régions par leurs surfaces externes. Les voisins d'un élément sont décrits dans le sens trigonométrique lorsqu'on observe la surface de l'extérieur, et décrivent des triangles.

Une autre représentation aurait été très efficace dans notre cas pour modéliser l'élasticité (comme nous le verrons au paragraphe 3) : les maillages simplex (Delingette, 1994). C'est une représentation duale de la triangulation (le passage d'une représentation à l'autre est toujours possible) où chaque élément n'a que trois voisins.

Lorsque l'on observe la surface de l'extérieur, les voisins sont décrits dans le sens trigonométrique afin de distinguer, au niveau de chaque élément, l'intérieur de l'extérieur de l'objet (voir Figure 2).

2.2 La dynamique du système

Une simulation basée sur la physique semble être une bonne solution pour modéliser des mouvements de manière naturelle. C'est pourquoi nous adoptons ce point de vue (voir chapitre 2). Notre modèle est construit sur une base physique. Pour cela les éléments géométriques se voient attribuer une masse (premier concept de la dynamique). C'est la solution la moins coûteuse en terme de calcul, mais aussi en terme de représentation et de compréhension.

Le deuxième concept de la dynamique est celui de force. Une force agit sur un objet au cours de la simulation pour modifier son évolution. Puisque les objets sont représentés par des éléments placés à leur surface, notre approche nous oblige à engendrer les forces au niveau de ces éléments.

Le paragraphe 2.2.1 décrit plus en détail les forces que nous utilisons, et notamment la force engendrée par un attracteur. Le paragraphe 2.2.2 formule le principe d'action-réaction lié aux forces. Le paragraphe 2.2.3 donne l'algorithme d'évolution classique pour intégrer ces forces au niveau des éléments et calculer leurs déplacements.

2.2.1 Les forces

Trois types de forces sont nécessaires à notre modélisation :

- **Champs de force** : cette force provient d'un champ de force. On connaît sa valeur pour l'élément considéré et à l'instant considéré. Cette valeur peut être une fonction de l'état de l'élément considéré (masse, cinétique). C'est le cas par exemple de la force gravitationnelle.
- **Force locale**. Une force dont on connaît le module et la direction est exercée sur un élément. C'est le cas par exemple d'une force induite par un autre élément (interaction) ou par un outil manipulé par l'utilisateur.
- **Force induite par un point d'attraction**. Dans ce cas on connaît seulement la nature de la force : un élément doit atteindre une position idéale minimisant une certaine fonctionnelle, ou satisfaisant un critère donné. Cette position idéale, ou point d'attraction, peut provenir d'un attracteur (fixe ou mobile) ou d'une position calculée en fonction de l'état courant.

Des trois types de force c'est la dernière qui nous intéresse le plus. C'est elle qui va nous permettre de modéliser l'élasticité, la motricité ou encore de définir des outils de contrôle de l'animation.

Deux choix sont possibles quant à la formulation de la force engendrée par un point d'attraction :

- **Formulation énergétique**. On exprime l'attraction par une énergie et on utilise le gradient de cette énergie comme force d'attraction. C'est généralement l'approche utilisée pour la segmentation d'images quand les zones d'attractions ne sont pas exactement définies (dans ce cas on utilise un objet déformable pour délimiter une région de l'image, voir des travaux sur les contours actifs : (Bainville, 1992), (Leitner, 1993) ou (Kass et al, 1988)).
- **Formulation directe**. Dans notre cas, les zones d'attractions sont clairement définies. On utilise alors une formulation directe en ajoutant directement un ressort multidimensionnel entre l'état courant de l'élément et la position de son attracteur. La loi d'évolution du ressort sera intégrée par les équations de la dynamique : la force d'attraction sera générée par le ressort. Le même principe est utilisé dans (Lamouret et al, 1995) comme aide à la définition de scénario de films d'animation (voir aussi (Lamouret, 1995)).

Pour modéliser différents comportements il faut introduire un poids sur chaque force. Cette pondération n'est pas toujours fixe et peut évoluer au cours du temps (par exemple la variation de la pondération d'une force de contraction permet de modéliser la variation d'activité musculaire). Deux comportements différents seront caractérisés par deux ensembles de pondérations des forces appliquées au système.

2.2.2 Principe d'action-réaction

Ce principe énonce que la somme des forces internes d'un système est toujours nulle. Lorsque deux éléments sont en interaction, la force d'interaction agissant sur l'un est l'inverse

de celle agissant sur l'autre. Ainsi, les forces internes ne peuvent déplacer le centre de gravité du système.

Dans notre modèle, lorsqu'une force d'interaction est générée, nous nous efforcerons de calculer la force de réaction exacte lui correspondant. Cependant cela n'est pas toujours possible : dans le cas d'une force engendrée par un attracteur calculé à partir de plusieurs éléments (cas de la mémoire de forme locale) aucune information n'est disponible pour la répartir sur les éléments participants. Dans ce cas on répartit la force de réaction sur l'ensemble des éléments de manière équipondérée.

2.2.3 Algorithme d'évolution

Le principe d'évolution que l'on applique, quand le système n'est pas contraint, est tiré directement de la discrétisation la loi fondamentale de la dynamique (voir paragraphe 1, chapitre 2). L'algorithme issu de ce principe est présenté à la Figure 3.

À chaque instant t de la simulation :

- Pour tout élément \mathbf{E}_i (de position \mathbf{p}_i^t) :
- Calculer les forces \mathbf{f}_i^t appliquées sur \mathbf{E}_i .
- Appliquer un schéma de discrétisation de type Euler :

$$m_i \ddot{\mathbf{p}}_i^{t+dt} = \mathbf{f}_i^t$$

$$\dot{\mathbf{p}}_i^{t+dt} = \dot{\mathbf{p}}_i^t + \ddot{\mathbf{p}}_i^{t+dt} dt$$

$$\mathbf{p}_i^{t+dt} = \mathbf{p}_i^t + \dot{\mathbf{p}}_i^{t+dt} \cdot dt$$
- Mettre à jour le nouvel état cinétique.

Figure 3. Algorithme d'évolution (sans contrainte). dt est le pas de temps utilisé pour la discrétisation de la loi fondamentale de la dynamique.

En utilisant cet algorithme dans le cas d'une force induite par un attracteur dynamique (voir Figure 4), le déplacement de l'élément le rapproche toujours du point d'attraction.

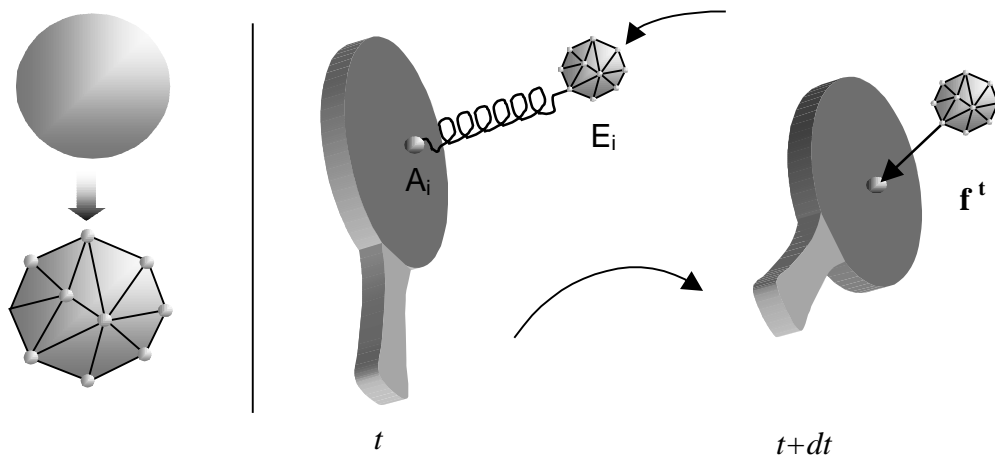


Figure 4. On utilise un ensemble d'éléments de contour pour définir la balle. Le point A_i est défini comme attracteur de l'élément E_i (de position \mathbf{p}_i). Ainsi quel que soit le mouvement de la raquette, l'élément E_i est attiré par le point A_i . La force générée par l'attracteur à chaque instant t est issue de la force d'un ressort liant l'élément et son attracteur : $\mathbf{f}^t = -k(\mathbf{p}_i^t - \mathbf{A}_i)$.

2.2.4 Récapitulatif sur les propriétés générales des éléments :

D'après ce qu'on a vu jusqu'à présent, un élément, de manière générale, est caractérisé par : un indice dans l'objet (unique), un numéro de région dans l'objet (unique), une position, un nombre de voisins, une liste de voisins, une masse, une vitesse, et la somme des forces qui lui sont appliquées.

Les régions solides ont ces mêmes propriétés, cependant les caractéristiques de voisinages ne sont utilisées que pour certains mode d'affichage (voir paragraphe 5).

Nous allons voir dans les prochains paragraphes comment, à partir d'une telle base, les différentes régions nécessaires à notre simulation sont modélisées : les régions élastiques, les régions musculaires et enfin les régions solides.

3. Régions élastiques

L'atout des modèles générateurs utilisant des principes physiques provient de leur aptitude à produire des comportements complexes et réalistes à partir d'une base et de règles simples. De nombreuses méthodes par assemblage d'éléments sont proposées dans la littérature. Cependant aucune ne satisfait complètement les contraintes suivantes :

- Facilité d'implémentation et faible besoin en calcul,
- Utilisation d'éléments surfaciques (et de facettes triangulées),
- Stabilité numérique,
- Réalisme des déformations.

Les deux derniers points sont très importants : le comportement élastique que nous voulons obtenir doit être très précis. Comme notre objectif est la modélisation des phases de la respiration, le système une fois déformé, doit retourner à sa configuration d'origine de manière précise. Cette dernière caractéristique est rarement rencontrée dans la littérature (souvent les retours sont partiels). Nous verrons que ce type de retour est difficile à obtenir (il demande à la fois une stabilité numérique et une précision importante).

Classiquement, le modèle le plus simple consiste à relier l'ensemble des éléments par des ressorts, les relations de voisinage étant retranscrites directement par un ressort. Mais cette approche, bien que très simple, n'est pas satisfaisante : le nombre important de ressorts rend le contrôle difficile.

Pour construire notre modèle, nous allons donc examiner deux formulations de l'élasticité définies dans la littérature de la dynamique : la théorie de l'élasticité et les tenseurs de déformations surfaciques. Nous en tirons les principes et présentons une définition originale de la forme à l'aide d'un repère local basé sur le voisinage (paragraphe 3.1.3). En appliquant cette définition, nous décrivons une mémoire de forme locale (paragraphe 3.2). Nous analyserons cette formulation de la déformation en la comparant avec une formulation classique masse-ressort (paragraphe 3.3).

3.1 Formulation de la forme et de la déformation

Sans reprendre ici l'ensemble de la théorie de l'élasticité (pour cela consulter (Ciarlet, 1985) ou (Bausset, 1982)), nous allons examiner deux façons d'exprimer une déformation : la formulation issue de la théorie de l'élasticité et celle utilisant les tenseurs surfaciques. Nous nous appuyons plus particulièrement sur la deuxième formulation pour définir une nouvelle formulation plus adaptée aux surfaces discrètes présentées au paragraphe 2.1.

3.1.1 Théorie de l'élasticité

La théorie de l'élasticité ne s'applique qu'en cas d'équilibre de l'objet considéré. L'évolution est quasi-statique et ne considère que les systèmes en état d'équilibre statique. La théorie est aussi valable pour des petites oscillations, mais ne convient pas à la modélisation de choc ou de déformation importante (cas de la collision). Cependant elle reste la seule base mathématique d'envergure dans ce cas là puisque aucune autre théorie physique n'est établie.

On considère un objet déformable O . Chaque point i (surfaccique ou non) de l'objet possède une position dans l'espace euclidien de dimension 3, notée \mathbf{x}_i . Sa configuration de référence est notée \mathbf{x}_i^0 , sa configuration à l'instant t considéré est \mathbf{x}_i^t . On connaît la fonction de déformation $F(\mathbf{x}_i)$ permettant d'amener l'objet de sa configuration d'origine jusque dans sa configuration présente.

On s'intéresse ici à la mesure locale de la déformation. On impose que la déformation préserve le sens et ne permet pas une compression infinie en un point, c'est-à-dire que $\nabla \mathbf{x}_i, \det(\nabla F(\mathbf{x}_i)) > 0$, où $\nabla F(\mathbf{x}_i)$ désigne le gradient de la fonction de déformation en \mathbf{x}_i (dans notre cas, c'est une matrice 3x3, correspondant à la matrice jacobienne de la fonction au point \mathbf{x}_i) :

$$\nabla F(\mathbf{x}_i) = \begin{pmatrix} \frac{\partial F^{(1)}(\mathbf{x}_i)}{\partial \mathbf{x}_i^{(1)}} & \frac{\partial F^{(1)}(\mathbf{x}_i)}{\partial \mathbf{x}_i^{(2)}} & \frac{\partial F^{(1)}(\mathbf{x}_i)}{\partial \mathbf{x}_i^{(3)}} \\ \frac{\partial F^{(2)}(\mathbf{x}_i)}{\partial \mathbf{x}_i^{(1)}} & \frac{\partial F^{(2)}(\mathbf{x}_i)}{\partial \mathbf{x}_i^{(2)}} & \frac{\partial F^{(2)}(\mathbf{x}_i)}{\partial \mathbf{x}_i^{(3)}} \\ \frac{\partial F^{(3)}(\mathbf{x}_i)}{\partial \mathbf{x}_i^{(1)}} & \frac{\partial F^{(3)}(\mathbf{x}_i)}{\partial \mathbf{x}_i^{(2)}} & \frac{\partial F^{(3)}(\mathbf{x}_i)}{\partial \mathbf{x}_i^{(3)}} \end{pmatrix}$$

On note $\mathbf{x}_i^{(j)}$ la $j^{\text{ème}}$ composante du vecteur \mathbf{x}_i (dans notre cas, j va de 1 à 3).

La matrice symétrique définie par :

$$\mathbf{C}(\mathbf{x}_i) = \nabla F(\mathbf{x}_i)^T \nabla F(\mathbf{x}_i)$$

est appelée le tenseur des déformations de Cauchy-Green.

Dans la théorie de l'élasticité, le tenseur des déformations de Green-Lagrange (appelé parfois de Green-St Venant), défini par :

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

où \mathbf{I} est la matrice identité, mesure l'écart entre la déformation produite par $F(\mathbf{x}_i)$ et un simple déplacement rigide. Nous retiendrons qu'il est nul lorsque la déformation est localement rigide (en \mathbf{x}_i).

On notera que les grandeurs de la théorie de l'élasticité sont définies localement pour tous les points de l'objet. Aucune mesure globale n'est étudiée. Le théorème de Cauchy permet de relier les forces appliquées à la surface et au volume de l'objet à la fonction de déformation.

Notre modèle étant surfaccique, nous allons mesurer les déformations à l'aide de la théorie des surfaces dans le paragraphe suivant.

3.1.2 Tenseurs de déformation surfacique

La mesure de la déformation de la surface de l'objet (son contour) doit être nulle pour tout déplacement rigide (invariance de l'objet par translation et rotation). Par contre, on doit avoir un résultat non nul pour toute déformation de l'objet par rapport à son état de repos. On ne peut pas utiliser une formule linéaire (qui serait directement dépendante des valeurs de la position du contour, c'est-à-dire une fonction linéaire des positions de ses éléments). On part de la même description que (Terzopoulos et al., 1987) : le théorème fondamental des surfaces énonce que deux surfaces ont la même forme si leurs tenseurs métriques et leurs tenseurs de courbures sont des fonction identiques des \mathbf{x}_i (point de surface). On associe ces deux valeurs à la déformation.

Nous examinons ces deux mesures pour une discrétisation de la surface par la méthodes des différences finies : la surface est paramétrée dans deux directions u_1 et u_2 . Une surface est donc définie par un ensemble de points $\mathbf{x}(u_1, u_2)$.

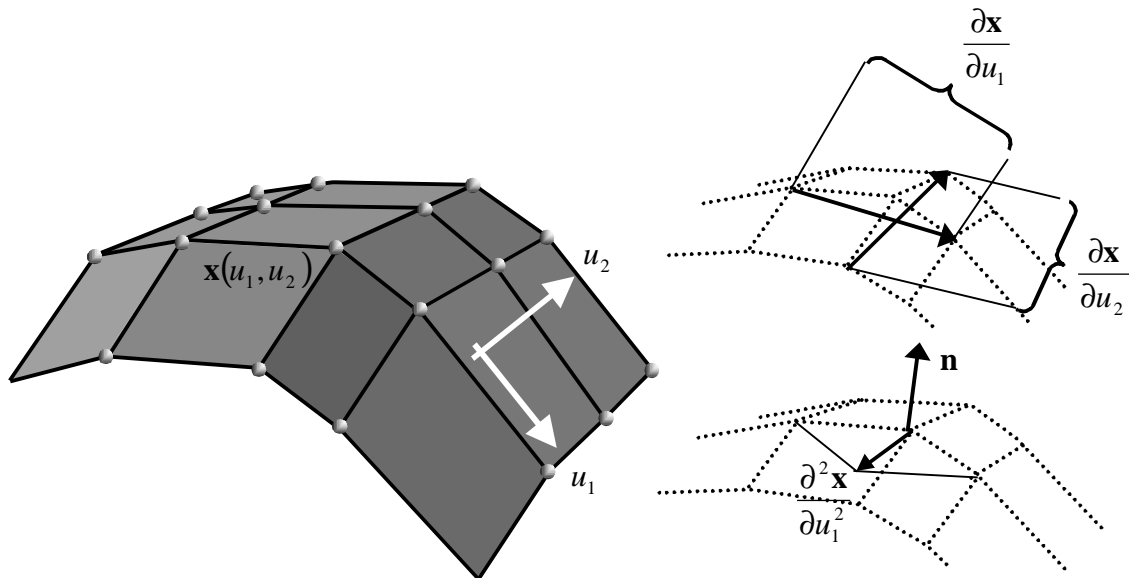


Figure 5. Tenseur métrique et tenseur de courbure. Les directions de paramétrisation sont u_1 et u_2 . La surface est représentée en plein sur la gauche.

a) Tenseur métrique

On mesure les longueurs (distances euclidiennes) entre les éléments voisins de la surface. Lorsque l'objet se déforme, ces distances changent. On cherche à obtenir une mesure de ces distances à partir desquelles nous pourrions déduire une force élastique de tension entre les éléments.

Dans le cas d'une surface 3D paramétrée, on définit le tenseur métrique par la matrice symétrique :

$$\mathbf{G}(\mathbf{x}(u)) = \begin{pmatrix} \left\langle \frac{\partial \mathbf{x}}{\partial u_1}, \frac{\partial \mathbf{x}}{\partial u_1} \right\rangle & \left\langle \frac{\partial \mathbf{x}}{\partial u_2}, \frac{\partial \mathbf{x}}{\partial u_1} \right\rangle \\ \left\langle \frac{\partial \mathbf{x}}{\partial u_1}, \frac{\partial \mathbf{x}}{\partial u_2} \right\rangle & \left\langle \frac{\partial \mathbf{x}}{\partial u_2}, \frac{\partial \mathbf{x}}{\partial u_2} \right\rangle \end{pmatrix}$$

Le tenseur métrique est représenté à la Figure 5.

b) Tenseur de courbure

Le tenseur métrique ne permet pas de déterminer entièrement la forme puisque la courbure peut changer sans changer la longueur des segments. On ajoute la mesure de courbure définie par la matrice symétrique suivante :

$$\mathbf{B}(\mathbf{x}(u)) = \begin{pmatrix} \left\langle \mathbf{n}, \frac{\partial^2 \mathbf{x}}{\partial u_1^2} \right\rangle & \left\langle \mathbf{n}, \frac{\partial^2 \mathbf{x}}{\partial u_1 \partial u_2} \right\rangle \\ \left\langle \mathbf{n}, \frac{\partial^2 \mathbf{x}}{\partial u_2 \partial u_1} \right\rangle & \left\langle \mathbf{n}, \frac{\partial^2 \mathbf{x}}{\partial u_2^2} \right\rangle \end{pmatrix}$$

Où \mathbf{n} est la normale à la surface (voir Figure 5).

Le tenseur métrique correspond à une mesure de la distance séparant les éléments voisins, le tenseur de courbure à une mesure de la courbure locale. Ces deux tenseurs suffisent pour décrire une surface tridimensionnelle. Cependant, dans le cas d'une courbe tridimensionnelle, il manque encore un tenseur pour l'information de torsion. Dans notre cas, cela ne nous intéresse pas.

Là encore, les tenseurs décrivent la forme locale de la surface.

Nous retenons que les deux tenseurs présentés ici mesurent une tension de surface (dérivée première) et une tension de courbure (dérivée seconde). Dans le paragraphe suivant nous allons formuler une nouvelle mesure de la déformation applicable directement sur les surfaces triangulées décrites au paragraphe 2.1 en regroupant les expressions des dérivées au sein d'une même formulation discrète.

3.1.3 Repère local basé sur le voisinage

Nous désirons trouver une expression de la forme locale. Dans notre approche par assemblage d'élément, on se place au niveau d'un élément et on cherche à définir chaque élément relativement à ses voisins. On cherche donc à exprimer la position \mathbf{x}_i d'un élément \mathbf{E}_i dans un repère local basé sur les voisins de \mathbf{E}_i .

On utilise pour cela trois paramètres en 3D (deux en 2D) pour chaque élément : deux angles et un coefficient. Cette définition comporte des cas singuliers que nous examinons en fin de paragraphe. Comme nous utilisons un vecteur « normal » à la surface en un point d'un maillage pour définir nos paramètres de forme, nous devons dans un dernier sous-paragraphe revenir sur cette notion de normale (voir aussi annexe 3).

a) *Définition des paramètres de forme*

À partir du vecteur \mathbf{n}_i normal à la surface en \mathbf{E}_i , et de l'isobarycentre \mathbf{G}_i des voisins de cet élément, on peut caractériser la position \mathbf{x}_i en prenant :

- α_i l'angle entre \mathbf{n}_i et $\mathbf{G}_i\mathbf{x}_i$.
- β_i l'angle entre le vecteur $\mathbf{u}_i = \mathbf{n}_i \times \mathbf{G}_i\mathbf{x}_i$ et le vecteur \mathbf{v}_i (\mathbf{v}_i est un vecteur normal à \mathbf{n}_i , nous verrons plus loin comment il est choisi).
- γ_i la distance entre l'isobarycentre \mathbf{G}_i et \mathbf{x}_i . C'est donc la longueur du vecteur $\mathbf{G}_i\mathbf{x}_i$.

L'élément \mathbf{E}_i est ainsi repéré par rapport à ses voisins en utilisant le triplet de trois scalaires $(\alpha_i, \beta_i, \gamma_i)$, voir Figure 6.

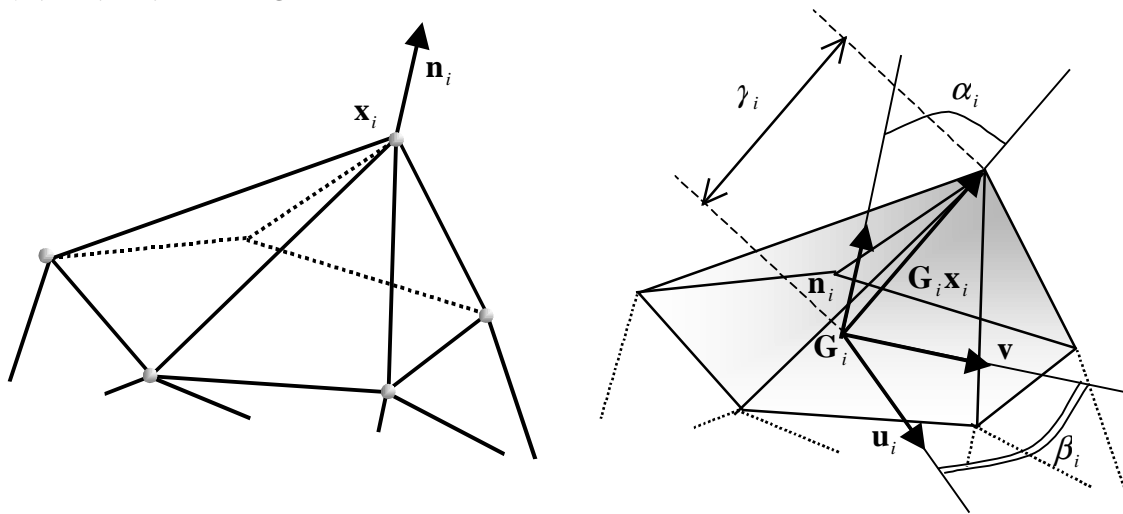


Figure 6 Repère local basé sur le voisinage.

Le vecteur \mathbf{v} est directement déduit de \mathbf{n}_i en utilisant $\mathbf{v}_i = (\mathbf{n}_i^{(2)}, -\mathbf{n}_i^{(1)}, 0)$.

L'avantage d'une telle représentation est qu'à partir des coordonnées des voisins et des trois paramètres on peut retrouver la position de l'élément.

Ce système de coordonnées locales est légèrement différent du système de coordonnées sphérique car il est basé directement sur la position des voisins (si les voisins se déplacent, \mathbf{G}_i , \mathbf{n}_i et donc \mathbf{v}_i changent). Cependant, si on considère des voisins fixes et que l'on fait varier les valeurs de α_i , de β_i et de γ_i on retrouve le même comportement que dans un repère sphérique classique (voir Figure 7).

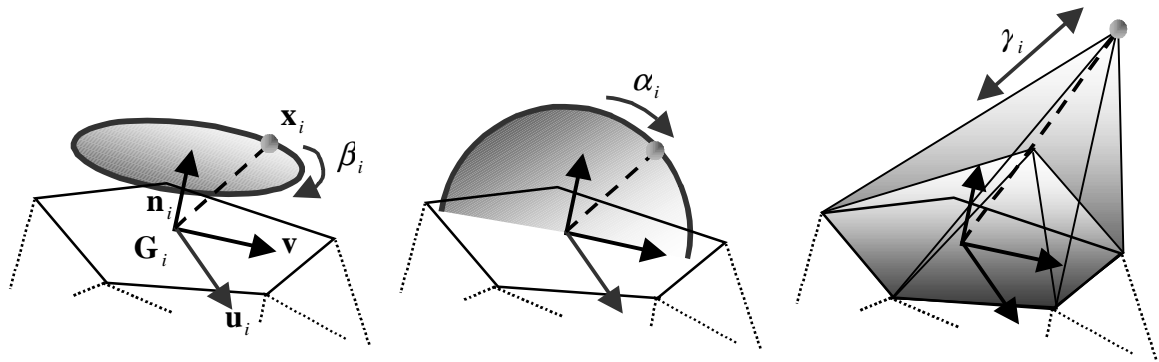


Figure 7. Étude des variations des paramètres de forme.

b) Cas particuliers

Il existe deux cas particuliers dans cette représentation :

- - $\mathbf{n}_i = (0, 0, 1)$: dans ce cas on ne peut pas prendre le même \mathbf{v}_i . On choisit donc de prendre $\mathbf{v}_i = (1, 0, 0)$.
- - $\alpha_i = \pm\pi/2$: alors \mathbf{x}_i est dans le plan de $\mathbf{G}_i\mathbf{u}_i$. On ne considère que l'angle β_i et la distance γ_i (on utilise un drapeau spécial indiquant si \mathbf{x}_i est confondu avec \mathbf{G}_i , c'est-à-dire si $\gamma_i = 0$).

c) Définition de la normale \mathbf{n}_i

La normale \mathbf{n}_i en un point \mathbf{x}_i d'une surface triangulée (donc discrète) n'existe mathématiquement pas. On cherche en fait une approximation de la normale en ce point de la surface approximée par les facettes. Cette approximation peut se faire de plusieurs façons équivalentes dans le cas de facettes triangulaires (voir Figure 8 et preuve en annexe 3) :

- Moyenne des normales des facettes concourantes en \mathbf{E}_i .
- Moyenne des produits vectoriels deux à deux des vecteurs $\mathbf{x}_i\mathbf{y}_i^j$ (où \mathbf{y}_i^j est le $j^{\text{ème}}$ voisin de l'élément \mathbf{E}_i).

On aura besoin de cette la valeur de la normale dans deux autres phases de la simulation : pour la contrainte d'incompressibilité et l'affichage (avec un rendu de Gouraud ou de Phong). Son calcul n'est donc pas coûteux.

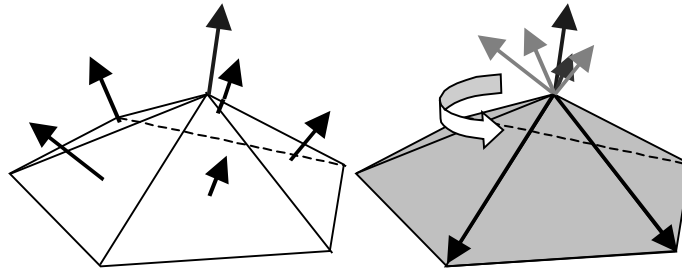


Figure 8. Deux façons de calculer \mathbf{n}_i . À gauche : moyenne des normales des facettes concourants en \mathbf{E}_i , au milieu moyenne des produits vectoriels deux à deux $\mathbf{x}_i \mathbf{y}_i^j \times \mathbf{x}_i \mathbf{y}_i^{j+1}$.

3.2 Mémoire de forme locale

Dans ce paragraphe nous montrons comment utiliser les paramètres définis ci-dessus et le principe de force engendré par un attracteur pour exprimer une mémoire de forme locale et par la même modéliser l'élasticité. Nous analyserons l'apport de notre méthode en le comparant à un modèle classique masse-ressort au paragraphe 3.3.

Pour exprimer l'élasticité du tissu on va utiliser le concept de force générée par un attracteur. Dans notre cas, l'attracteur est la position idéale qui satisfait au mieux la forme locale de départ, pour un élément \mathbf{E}_i on le note \mathbf{a}_i .

Pour cela :

- On mémorise les paramètres α_i , β_i et γ_i lorsque l'objet est dans sa position de repos. Cela peut être fait à tout moment de la simulation ce qui permet d'utiliser la simulation comme outils de modelage d'objet. En partant d'un objet initial et en le déformant sous l'action de force ou de contraintes, on obtient un nouvel objet dont on mémorise les paramètres de forme. Notons α_i^0 , β_i^0 et γ_i^0 les paramètres de la forme de repos.
- On calcule, à tout instant de la simulation, pour un élément \mathbf{E}_i , la position de son attracteur. À partir de la position des voisins et des paramètres de la forme de repos, on calcule \mathbf{G}_i , et \mathbf{n}_i , on en déduit \mathbf{v}_i . Par rotation autour de \mathbf{n}_i d'un angle de $-\beta_i^0$ on trouve \mathbf{u}_i . Par rotation autour de \mathbf{u}_i d'un angle de $-\alpha_i^0$, on trouve la direction de $\mathbf{G}_i \mathbf{a}_i$. Ce vecteur est multiplié par γ_i^0 . On en déduit la position de l'attracteur \mathbf{a}_i (notons qu'à la forme de repos $\mathbf{a}_i = \mathbf{x}_i$)
- On simule une force d'attraction entre \mathbf{x}_i (position actuelle de \mathbf{E}_i) et \mathbf{a}_i en modélisant un ressort entre les deux positions. Une force $\mathbf{f} = -k(\mathbf{x}_i - \mathbf{a}_i)$ est créée. On utilise la discrétisation des lois de la dynamique pour connaître la force élastique exercée sur \mathbf{x}_i à cette itération.

Ainsi, à chaque itération, on doit calculer pour chaque élément la position de l'attracteur \mathbf{a}_i . En terme de calcul, cela représente la création de deux matrices de rotation, deux multiplications matrice-vecteur, une soustraction de vecteurs et deux multiplications scalaire-vecteur.

3.3 Analyse de l'apport de la mémoire de forme

Nous allons étudier le système composé par un objet élastique dont les propriétés élastiques sont exprimées par la mémoire de forme. La preuve théorique de l'existence d'une solution convergente lorsqu'un tel système, écarté de sa forme de repos puis relaxé, n'a pas été étudiée au cours de cette thèse. Afin d'étudier cette convergence et d'évaluer l'apport de notre concept, nous avons choisi de comparer notre modèle à un modèle masse-ressort classique en utilisant le même réseau d'éléments de masses reliées par des facettes triangulaires. L'élasticité du modèle masse-ressort est modélisée en remplaçant chaque arête de facette par des ressorts. Cela comporte deux avantages :

- la comparaison des méthodes sur les mêmes objets,
- l'implémentation diffère seulement dans le calcul de la force d'élasticité.

Nous allons effectuer deux types de comparaison : une comparaison formelle en 2D en utilisant les méthodes classiques d'analyse de stabilité (Edelstein-Keshet., 1988), et une comparaison empirique en simulant des perturbations sur deux objets 3D identiques. Cette étude a été présentée au congrès Conference on Dynamical Systems in Biology and Medicine, Veszprém (Hongrie) en juillet 1996 (« a physically-based model for simulation of respiratory movements », E. Promayon, P. Baconnier, C. Puech).

3.3.1 Comparaison formelle

La comparaison formelle consiste à comparer analytiquement la stabilité des systèmes. On pose dans un premier temps les bases théoriques (a), puis nous les appliquons à notre étude (b). Nous choisissons enfin un exemple pratique simple nous permettant de tirer des conclusions sur la comparaison formelle (c).

a) Bases théorique (Edelstein-Keshet., 1988)

Soit un système quelconque discret, on note \mathbf{x}^t le vecteur d'état au temps t . La dimension de \mathbf{x} est égale au nombre d'éléments du système multiplié par le nombre de dimensions considérées (en général on utilise la position dans l'espace euclidien comme état, le nombre de dimensions étant 2 ou 3). On considère la fonction d'évolution globale en posant :

$$\mathbf{y}^{t+dt} = f(\mathbf{y}^t) \text{ avec } \mathbf{y}^t = \begin{pmatrix} \mathbf{x}^t \\ \dot{\mathbf{x}}^t \end{pmatrix}.$$

La stabilité d'un tel système discret, au voisinage du point \mathbf{y}^t peut être étudiée grâce à sa matrice jacobienne :

$$\mathbf{J}^{(i,j)} = \frac{\partial f_i(\mathbf{x})}{\partial \mathbf{y}_j},$$

c'est-à-dire que l'élément (i, j) de la matrice jacobienne est la dérivée partielle de la fonction d'évolution de la $i^{\text{ème}}$ composante par rapport à la $j^{\text{ème}}$ composante du vecteur d'état.

En étudiant la matrice jacobienne, on étudie donc la variation de la fonction d'évolution par rapport à la variation de l'état. Cela permet de caractériser la fonction d'évolution. Dans le cas d'un système discret (dans le temps), le système est stable lorsque :

$$\forall \lambda_i, \quad \text{si } \lambda_i \text{ est réelle alors } |\lambda_i| < 1, \\ \text{si } \lambda_i \text{ est complexe alors } |\lambda_i| < 1 \text{ et } \operatorname{Re}(\lambda_i) \neq 0,$$

où les λ_i sont les valeurs propres de la matrice jacobienne et $\operatorname{Re}(\lambda_i)$ est la partie réelle de λ_i . On peut interpréter ce résultat en notant qu'à chaque itération on restreint le nombre d'évolutions possibles : il y a convergence.

b) Application

Dans notre cas, la fonction d'évolution est complètement liée à la fonction d'élasticité. On va comparer la stabilité de la fonction d'évolution en utilisant des systèmes « masse-ressort » et « mémoire de forme locale ».

Ces deux systèmes suivent des approches par assemblage d'éléments : on va intégrer les lois de la dynamique sur l'ensemble des n éléments les composants.

Les systèmes étudiés sont discrets :

- discrets dans l'espace puisque constitués d'éléments connus,
- discrets dans le temps puisqu'on passe d'un état au temps t à un état au temps $t + dt$.

On applique directement la méthode formelle décrite au paragraphe (a).

La description des deux systèmes à comparer se déroule en deux étapes : description des éléments (points de masse) et description des forces appliquées. On obtient la fonction d'évolution.

Les éléments, points de masse :

Soit \mathbf{x} le vecteur de taille $3n$ contenant l'ensemble des positions des éléments au temps t , on a $\mathbf{x}^t = (\mathbf{x}_1^{(1)t}, \mathbf{x}_1^{(2)t}, \mathbf{x}_1^{(3)t}, \dots, \mathbf{x}_n^{(3)t})$. Chaque élément du système a une masse m .

Les forces :

Deux forces sont modélisées :

- la force d'élasticité F^e (qui dépend de la position au temps t et du modèle utilisé), elle s'exprime différemment dans les deux systèmes, comme nous verrons plus loin.
- la force de viscosité (qui dépend de la vitesse au temps t , et qui est identique pour les deux modèles). Les forces de viscosité présentes dans les modèles générateurs sont à la fois utilisées pour le réalisme physique, mais aussi pour l'amélioration de la convergence.

Dans les deux systèmes on a :

$$\mathbf{F}^v = -k_v \cdot \dot{\mathbf{x}}^t$$

La fonction d'évolution :

La loi fondamentale de la dynamique nous donne :

$$\sum \mathbf{f} = \mathbf{F}^e + \mathbf{F}^v = m\ddot{\mathbf{x}}$$

On exprime le schéma d'Euler comme un système d'équations :

$$\mathbf{y}^{t+dt} = f(\mathbf{y}^t) \Leftrightarrow \begin{cases} \mathbf{x}^{t+dt} = \mathbf{x}^t + dt \cdot \dot{\mathbf{x}}^t \\ \dot{\mathbf{x}}^{t+dt} = \dot{\mathbf{x}}^t + dt \cdot \left(\frac{\mathbf{F}^e(\mathbf{x}^t) + \mathbf{F}^v(\dot{\mathbf{x}}^t)}{m} \right) \end{cases}$$

Pour simplifier l'écriture on pose $k_v = \frac{k_v}{m}$, on obtient :

$$\begin{cases} \mathbf{x}^{t+dt} = \mathbf{x}^t + dt \cdot \dot{\mathbf{x}}^t \\ \dot{\mathbf{x}}^{t+dt} = \dot{\mathbf{x}}^t + dt \cdot \left(\frac{\mathbf{F}^e(\mathbf{x}^t)}{m} - k_v \cdot \dot{\mathbf{x}}^t \right) \end{cases}$$

Les deux méthodes ne diffèrent que dans le calcul de \mathbf{F}^e .

c) Exemple pratique

Nous verrons dans ce paragraphe l'exemple simple que nous avons choisi. À partir de cet exemple nous expliciterons la matrice jacobienne commune, la force élastique (différentes pour les deux modèles). Avant d'obtenir les résultats de la comparaison, nous allons devoir trouver l'équivalence des poids des forces élastiques.

Énoncé

Pour continuer l'étude, on se place dans un cas très simple (voir Figure 9) : on utilise une seule facette triangulaire **ABC** située dans le plan $z=0$ (donc il s'agit d'un cas bi-dimensionnel) dont deux éléments sont fixes (**B** et **C**). On étudie la stabilité de l'élément libre **A** autour de son point d'équilibre (dans notre cas l'état d'équilibre est atteint lorsque **A** est revenu à sa position de repos).

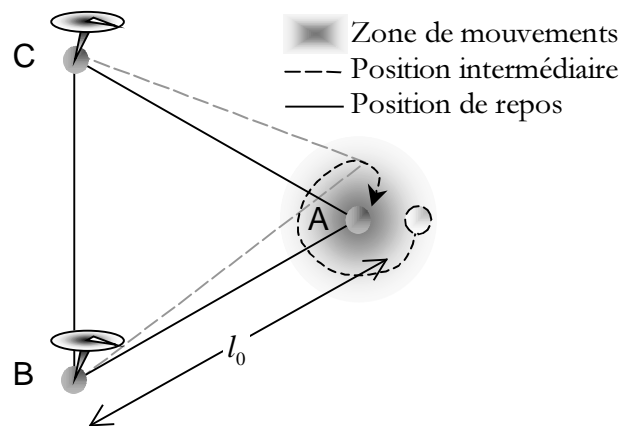


Figure 9. Comparaison de stabilité. On perturbe la position de l'élément **A** et on observe son retour en position d'équilibre (les deux autres éléments **B** et **C** sont fixes). l_0 est la longueur des arêtes à la position de repos.

Matrice jacobienne

Dans ce cas, on peut expliciter la matrice jacobienne du système décrit au paragraphe (b). Dans la matrice suivante on ne présente que les lignes et colonnes correspondants au point \mathbf{A} (les autres sont toutes nulles) et seulement pour les deux dimensions significatives ; les abscisses sont en lignes 1 et 3 et en colonne 1 et 3, la position en lignes 1 et 2 et colonnes 1 et 2 :

$$\mathbf{J} = \begin{pmatrix} 1 & 0 & dt & 0 \\ 0 & 1 & 0 & dt \\ -\frac{\partial F_{\mathbf{x}^{(1)}}^e}{m \partial \dot{\mathbf{x}}^{(1)}} & -\frac{\partial F_{\mathbf{x}^{(2)}}^e}{m \partial \dot{\mathbf{x}}^{(1)}} & 1 - \frac{dt \cdot k_v}{m} & 0 \\ -\frac{\partial F_{\mathbf{x}^{(1)}}^e}{m \partial \dot{\mathbf{x}}^{(2)}} & -\frac{\partial F_{\mathbf{x}^{(2)}}^e}{m \partial \dot{\mathbf{x}}^{(2)}} & 0 & 1 - \frac{dt \cdot k_v}{m} \end{pmatrix}$$

Forces élastiques

La fonction F^e diffère selon les modèles, les éléments \mathbf{B} et \mathbf{C} étant fixes, on ne s'intéresse qu'à la force élastique générée sur l'élément \mathbf{A} :

- **système masse-ressort** : les couples (\mathbf{A}, \mathbf{B}) , (\mathbf{B}, \mathbf{C}) et (\mathbf{A}, \mathbf{C}) sont reliés par des ressorts. Les éléments \mathbf{B} et \mathbf{C} étant fixes, le ressort (\mathbf{B}, \mathbf{C}) n'est pas modélisé. Restent les ressorts (\mathbf{A}, \mathbf{B}) et (\mathbf{A}, \mathbf{C}) . On obtient, en \mathbf{A} :

$$F^e = k_1 \cdot \left(1 - \frac{l_0}{\|\mathbf{AB}^t\|} \right) \mathbf{AB}^t + k_1 \left(1 - \frac{l_0}{\|\mathbf{AC}^t\|} \right) \mathbf{AC}^t$$

où \mathbf{AB}^t est le vecteur \mathbf{AB} à l'instant t . k_1 est le coefficient de la force élastique dans le cas du système masse-ressort.

- **mémoire de forme locale** : dans ce cas, seul reste le ressort entre l'élément \mathbf{A} et son attracteur. La position de l'attracteur est fixe car, comme on l'a vu au paragraphe 3.2, elle est définie à l'aide de la position des voisins de \mathbf{A} : \mathbf{B} et \mathbf{C} , ces derniers étant fixes, l'attracteur sera toujours en \mathbf{A}^0 . On exprime donc la force élastique par :

$$F^e = k_2 (\mathbf{A}^t - \mathbf{A}^0)$$

k_2 est le coefficient de la force élastique dans le cas de la mémoire de forme locale.

Équivalence des poids k_1 et k_2

Afin de comparer de manière équivalente les deux systèmes, il faut trouver une équivalence entre les constantes de raideur k_1 et k_2 . Pour cela on déplace l'élément \mathbf{A} dans une direction donnée de telle sorte que les directions des forces soient les mêmes puis on cherche k_1 et k_2 tels que les forces élastiques aient les même modules (voir Figure 10). La direction satisfaisant la première contrainte est perpendiculaire à l'arête \mathbf{BC} . Dans ce cas là les deux forces élastiques sont colinéaires.

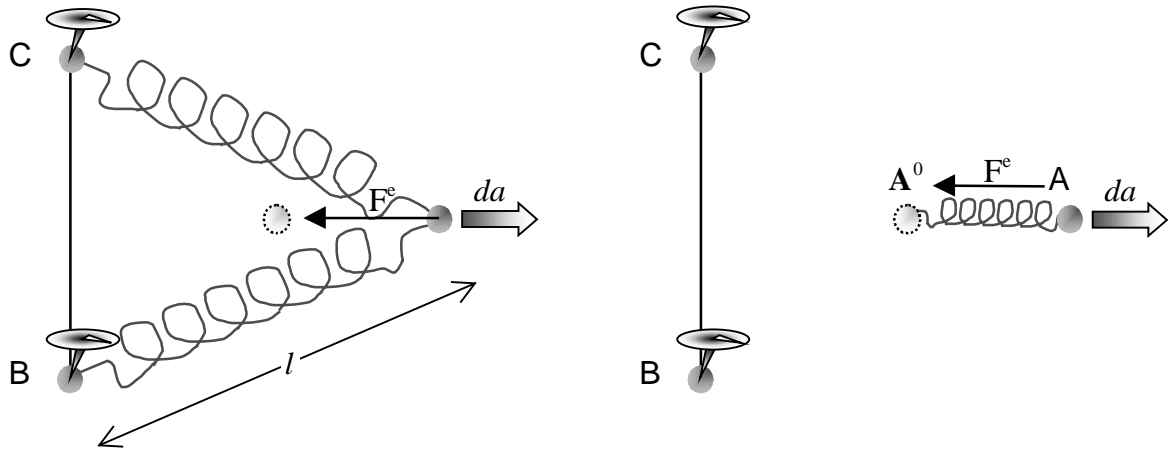


Figure 10. Méthode de calibrage des constantes de raideur. À gauche cas du système masse-ressort, à droite de la mémoire de forme locale.

Soit da la distance de déplacement de l'élément **A** le long de cet axe. On calcule le coefficient c tel que $k_2 = c \cdot k_1$ en comparant les modules des deux forces. On obtient c en fonction de da :

$$c(da) = (l - l_0) \frac{2 \cdot da + l_0 \sqrt{3}}{l \cdot da},$$

où $l = \sqrt{l_0^2 + da^2 + l_0 \cdot da \sqrt{3}}$ est la norme du vecteur **AB** et du vecteur **AC**.

L'étude de la fonction $c(da)$ montre que :

$$\lim_{da \rightarrow 0} c(da) = \frac{3}{2}$$

Comme l'étude de stabilité n'est valable que pour de petites perturbations (petits écarts par rapport à la position d'équilibre), c'est bien cette limite que nous allons utiliser.

Résultats

On exprime les valeurs propres de la matrice jacobienne en fonction d'un coefficient k unique (égal à k_2), et du pas de temps dt :

- système masse ressort : $\left\{ 1 - \frac{k_v}{m} dt \quad 1 - \frac{k_v \pm \sqrt{k_v^2 - \frac{16}{3} m \cdot k}}{2m} dt \right\}$
- mémoire de forme locale : $\left\{ 1 - \frac{k_v \pm \sqrt{k_v^2 - 4m \cdot k}}{2m} dt \right\}$

On recherche les valeurs limites du pas de temps et de k , telles que les valeurs propres respectent la convergence. On obtient le Tableau 1.

	Système masse-ressort	Mémoire de forme locale
Valeurs limites du pas de temps dt	$\frac{m}{k_v}$	$2 \frac{m}{k_v}$
Valeurs limites de la constante de raideur k	$\frac{3}{4} \cdot \frac{m}{4}$	$\frac{m}{4}$

Tableau 1. Valeurs limites des paramètres de stabilité du système (constante de raideur et pas de temps).

D'une part, le système masse-ressort supporte des pas de temps inférieurs à ceux supportés par la mémoire de forme locale puisque dans ce dernier on peut doubler le pas de temps sans diverger. D'autre part, la mémoire de forme locale offre une plus large palette de comportement puisqu'on peut utiliser des constantes de raideurs plus élevées sans divergence (donc modéliser des objets plus rigides).

La comparaison est poursuivie au paragraphe suivant par une comparaison empirique : un objet plus complexe est simulé en utilisant les deux modèles d'élasticité.

3.3.2 Comparaison empirique : simulation

La comparaison formelle n'est pas complètement satisfaisante pour trois raisons :

- La complexité de l'étude théorique nous oblige à utiliser un cas 2D avec un objet simple. Ce n'est pas ce qui nous intéresse ici puisque nous manipulons toujours des objets tridimensionnels plus complexes.
- D'un côté, elle est plus avantageuse pour la mémoire de forme locale : en fixant 2 points on supprime deux ressorts, alors que dans le système masse-ressort seul le ressort (**B**, **C**) est supprimé. On retrouve ce résultat directement sur la valeur limite du pas de temps applicable (voir Tableau 1).
- De l'autre, elle ne prend pas en compte tous les avantages de la mémoire de forme. En effet, dans le système masse-ressort simple utilisé ici, seules les distances entre les éléments sont modélisées. La convexité n'est pas prise en compte. Il existe des configurations ambiguës (voir Figure 11).

Pour éviter cet inconvénient il est nécessaire d'ajouter des ressorts angulaires (pour mémoriser la convexité). En ajoutant au moins un ressort supplémentaire par élément, le système masse-ressort est plus complexe et sa stabilité globale diminue. Ce n'est pas le cas avec la mémoire de forme locale.

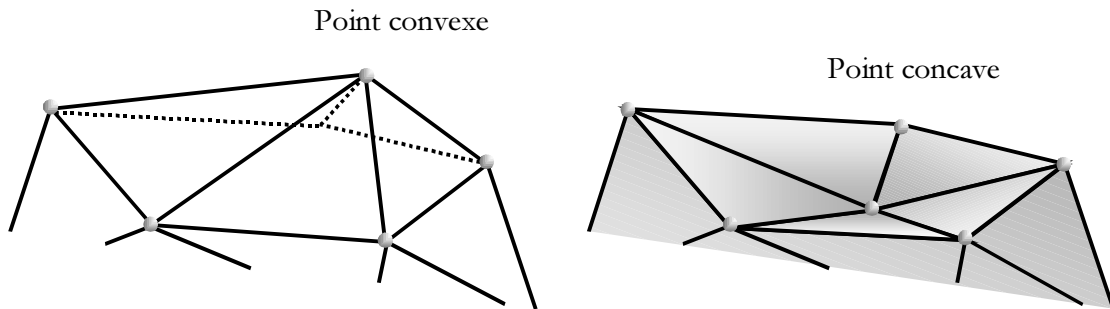


Figure 11. Ambiguïté du système masse-ressort simple (celui utilisé dans la comparaison). Dans les deux cas ci-dessus le système masse ressort modélise l'objet de la même façon.

Nous avons donc décidé de comparer directement des simulations sur un objet 3D : un icosaèdre (12 éléments, 20 facettes triangulaires). On perturbe l'objet, en le déformant par rapport à sa position de repos, et on observe son retour à la forme de référence. En utilisant la même méthode de calibrage des constantes de raideurs qu'au paragraphe précédent, on étudie, pour une élasticité donnée et une amplitude de perturbation donnée, la valeur limite du pas de temps. Cette valeur limite du pas de temps est la valeur au-delà de laquelle l'objet ne parvient pas à revenir à son état d'équilibre (c'est-à-dire qu'il lui faut plus de \max_i itérations pour retourner à son état d'équilibre). L'amplitude de la perturbation est mesurée en fonction de la longueur l_0 des arêtes des faces au repos. Ainsi une perturbation de $p = 0,5$ signifie que chaque élément a été déplacé dans une direction non-isotropique (tirée au hasard dans les 3 dimensions) et d'amplitude $0,5 \cdot l_0$.

Pour les expérimentations, on a fixé le seuil du nombre d'itérations à $\max_i = 5000$ et on a pris deux types de perturbations : des perturbations de faible amplitude ($0,3 \cdot l_0$) et de grande amplitude ($0,7 \cdot l_0$). Les résultats sont données à la Figure 12 et au Tableau 2.

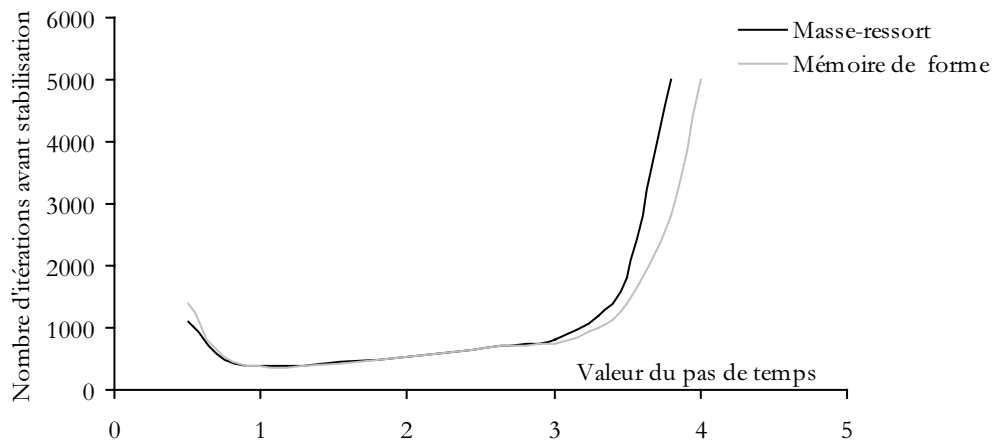
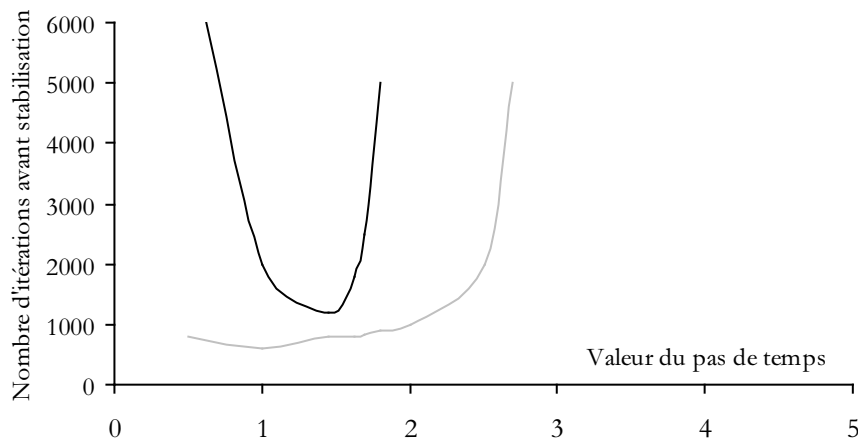
a) Faibles perturbations ($p=0,3$)b) Fortes perturbations ($p=0,7$)

Figure 12. Simulation comparée d'un icosaèdre déformé de sa configuration d'équilibre dans le cas de faible (a) et de forte (b) perturbations.

	Système masse-ressort	Mémoire de forme locale
Faibles perturbations ($p=0,3$)	3,8	4,0
Faibles perturbations ($p=0,7$)	1,8	2,7

Tableau 2. Valeurs limites du pas de temps avant divergence.

Les valeurs limites du pas de temps (Figure 12) montrent que dans le cas de petites perturbations les deux modèles sont sensiblement équivalents. Le rappel des ressorts (qui tendent simplement à rétablir des distances) fonctionne correctement car la forme géométrique n'a que peu changé. Par contre, dans le cas de fortes perturbations, la forme change beaucoup et le système de ressort ne suffit pas pour retrouver la géométrie de repos. De plus, dans certains cas, l'équilibre est atteint mais la forme obtenue n'est pas celle de départ (singularités identiques à celles présentée à la Figure 11).

3.3.3 Conclusion

Pour conclure sur la comparaison de notre modèle d'élasticité par rapport au modèle masse-ressort classique, plusieurs points sont à prendre en compte :

- Les modèles sont sensiblement équivalents quand la déformation est un changement d'échelle (le rappel des ressorts se faisant simplement sur les distances, la convergence est obtenue facilement) ;
- Le modèle de mémoire de forme locale paraît plus sensible aux erreurs de calculs (à cause des opérations de rotation de vecteurs). Cependant, il est plus stable dans le cas de déformations importantes. La robustesse provient du fait que la forme est modélisée explicitement et que le nombre de ressorts est moindre (seulement n ressorts instantanés).
- Le nombre de ressorts est toujours égal au nombre d'éléments pour la méthode de mémoire de forme. Plus les objets sont complexes plus la stabilité relative est importante. De plus, une contrainte fixant un élément dans l'espace euclidien diminue toujours le nombre de ressorts à modéliser (alors qu'il faut qu'un couple d'éléments soit fixé pour enlever un ressort du modèle masse-ressort).
- Le principal avantage du modèle masse-ressort reste calculatoire. La méthode de mémoire de forme est plus gourmande en temps de calcul.

Enfin, le Tableau 3 présente les caractéristiques du modèle suivant les critères donnés au chapitre précédent.

Principe de loi d'évolution	Manipulation de la forme	Forme de repos	Opérateur de différence
Intégration de la loi fondamentale de la dynamique	Éléments et paramètres de forme α_i , β_i et γ_i	Paramètres α_i^0 , β_i^0 et γ_i^0	Point d'attraction (paramètres de forme)

Tableau 3. Classement de la méthode de mémoire de forme selon les critères définis au chapitre 2.

3.4 Récapitulatif sur les éléments et régions élastiques

Un élément d'une région élastique hérite de toutes les propriétés d'un élément de type général (présentées au paragraphe 2.2.4). De plus, il a des caractéristiques spécifiques qui sont les paramètres de forme α_i^0 , β_i^0 et γ_i^0 , et le poids de la force de mémoire de forme (c'est-à-dire la constante de raideur associé au ressort qui le lie à son attracteur).

4. Régions musculaires

Un cas particulier des régions élastiques est indispensable à la modélisation des mouvements de la respiration. Il s'agit des régions musculaires. Ces régions ont la capacité de se contracter sous l'action d'une force et de revenir à leur configuration initiale lorsque cette force est annulée.

Nous allons examiner les différentes solutions offertes dans la littérature. Cela nous permettra, comme dans le cas des modèles élastiques, de définir l'approche s'appliquant le mieux à nos objectifs. Nous avons choisi de présenter l'état de l'art ici plutôt qu'au chapitre précédent car il est beaucoup plus spécifique et plus court que l'état de l'art sur les modèles déformables du chapitre précédent.

4.1 État de l'art

Pour simuler les mouvements de personnages les muscles sont souvent nécessaires et apportent beaucoup de réalisme. Deux approches principales ont été développées : la première s'attache à la visualisation des déformations induites par les muscles pour l'animation de personnages synthétiques, la seconde se veut plus réaliste et simule le fonctionnement réel des muscles.

4.1.1 Simulation des déformations

Dans ce contexte, l'important est de visualiser les déformations dues aux actions musculaires. Le raisonnement est inverse au fonctionnement réel : ici, le mouvement et la position du squelette (sur lequel sont attachés les muscles) sont répercutés sur les muscles, créant ainsi leurs déformations.

On peut citer les méthodes développées par (Chadwick et al., 1989), (Volino et al, 1996) et (Wilhelms, 1997). Le but du travail de (Chadwick et al., 1989) est une modélisation hiérarchique des personnages animés. La modélisation des muscles est basée sur des déformations de formes libres (FFD). À partir de la géométrie et de la cinématique des segments et des jointures du squelette portant les muscles, les déformations sont calculées sur les FFD représentant les muscles. La déformation visuelle est directement décrite par les FFD.

Dans (Volino et al, 1996) l'approche est aussi une approche hiérarchique multicouches. Les personnages sont découpés en trois couches : le squelette, l'enveloppe corporelle (dont les muscles) définie par des surfaces implicites (« metaballs ») associées aux primitives du squelette et la peau. En déplaçant le squelette ou en déformant les surfaces implicites les déformations musculaires peuvent être visualisées. La même approche est suivie dans (Wilhelms, 1997). Le but est de modéliser les animaux à partir de leur anatomie (os, muscle, et

tissus mous). Ce modèle donne de bon résultat tant pour la génération d'un modèle d'animal qu'au niveau visuel. La peau est recalculée à chaque itération à partir des mouvements du squelette et des tissus mous (qui ne sont en fait pas déformés) et de la déformation des tissus musculaires. Chaque élément anatomique est représenté par une ellipse. La déformation des muscles est simplement déduite de la position des ellipses du squelette sur lequel elles sont attachées et du contrôle des proportions et du volume. Il n'y a ni fonction d'activation, ni force motrice.

Dans ces approches, les muscles n'engendrent pas le mouvement, mais leurs déformations sont créées à partir du mouvement. Dans notre modélisation, nous voulons utiliser les muscles pour déplacer et déformer les composants du tronc, et cela pour deux raisons :

- dans le cas des mouvements respiratoires, et à l'inverse des mouvements décrits précédemment, les modifications de la forme extérieure du corps ne sont pas déductibles facilement à partir des mouvements du squelette.
- Dans le cadre de nos objectifs, à l'opposé des articles présentés, c'est un modèle fonctionnel et comportementale que nous désirons obtenir.

On doit donc rejeter ces méthodes.

4.1.2 Simulation du fonctionnement

Les méthodes qui simulent non pas les déformations mais le fonctionnement des muscles se rapprochent plus de notre approche et sont de deux types : les modèles qui considèrent le muscle comme un ou un ensemble de segment(s) et celles qui utilisent directement la forme du muscle.

a) Simulation par segments

Il s'agit de la méthode développée dans (Waters, 1987) et étendue dans (Lee et al., 1995). Dans (Waters, 1987), le but est la modélisation des muscles du visage pour l'animation faciale. On utilise le concept de FACS (Facial Action Coding System) qui donne une base pour déterminer les expressions liées aux émotions. Les muscles sont groupés par unité d'action. Un champ de forces est appliqué (combinaison du vecteur représentant le muscle et d'une zone d'influence) sur les ensembles de masses d'un maillage représentant le visage. Deux types de muscles sont modélisés : les muscles droits et les muscles sphincters (pour les yeux et la bouche). (Lee et al., 1995) améliorent la technique précédente en modélisant le visage par trois couches : une couche interne (les os), une couche intermédiaire et une couche externe (la peau) et en utilisant un modèle déformable discret (approche par assemblage d'éléments). Les muscles sont toujours modélisés par des vecteurs. Ils sont ancrés à la couche interne mais se situent au niveau de la couche intermédiaire. Ils sont soit constitués par une seule « fibre » (un seul vecteur), soit de plusieurs fibres (plusieurs segments). Ils génèrent des forces de déformations sur certains nœuds de la couche intermédiaire (déformés en fonction de leur placement par rapport aux fibres).

Dans (Beylot et al, 1996) et (Maurel et al, 1996), les muscles sont modélisés par un segment (ligne d'action principale du muscle) et des points d'attachement. Les muscles ont une masse, une force, les paramètres d'activation étant définis sur les segments. À chaque muscle correspond une zone physiologique de type cylindrique. Ces deux travaux s'insèrent dans le projet européen CHARM dans lequel les muscles seront à l'avenir modélisés par la méthode des éléments finis.

Dans (Miller, 1988), les muscles moteurs sont définis par des ressorts entre les composants du corps du serpent.

Dans les méthodes ci-dessus, les forces générées par les muscles induisent un mouvement sur les éléments du squelette reliés aux muscles. Cependant, la forme du muscle est représentée par un ou plusieurs segments. Ainsi, seuls les muscles plats et ayant des points d'attachements simples sur le squelette peuvent être modélisés.

b) Simulation de la forme

Le modèle de muscle le plus complet est celui de D. Chen (voir (Chen, 1992) et (Chen et Zeltzer, 1992)). À partir d'une étude fine de l'anatomie et du fonctionnement des muscles (notamment des fonctions de contraction, voir chapitre 1), et du modèle des forces musculaires de Zajac, les forces musculaires sont définies sous forme d'équations. La géométrie des muscles est définie sous forme polyédrique en segmentant des coupes IRM. On utilise la méthode des éléments finis (éléments de 20 nœuds) pour calculer la réponse élastique des muscles (donc la déformation). Des ressorts modélisent les forces musculaires et les forces des tendons. Ces ressorts sont attachés aux éléments finis afin de transmettre les forces. Cette modélisation tient compte des propriétés des fibres musculaires et des tendons qui les relient au squelette. La densité, le module d'Young et le coefficient de Poisson des tissus musculaires sont choisis à partir de mesures physiologiques. Les courbes de tension-longueur sont calculées à partir d'une simulation d'un muscle de grenouille et les résultats obtenus sont satisfaisants (bien que dans le cas de relaxation rapide du muscle, on voit apparaître des instabilités). Dans cette méthode, les éléments finis ne sont pas la source des forces musculaires, mais sont utilisés dans le calcul des déformations. De plus, la gravité n'est pas simulée. Enfin, les forces musculaires induites sur le squelette ne sont pas clairement définies.

J.C. Lombardo (Lombardo, 1996) et (Lombardo et Puech, 1995), utilise des particules orientées pour modéliser les muscles. Leur forme est donc représentée par un système de particules. Des lois d'interactions entre les particules permettent de créer et de simuler un comportement musculaire (interaction de conservation de volume notamment).

4.1.3 Conclusion

Nous avons vu que les méthodes conçues spécifiquement pour simuler les déformations des tissus musculaires ne correspondent pas à notre optique. D'autres part, les méthodes qui ne modélisent pas la forme directement ne peuvent pas convenir non plus. L'approche la plus utile est celle qui utilise la géométrie du muscle pour générer à la fois la déformation des tissus musculaires et les forces génératrices de mouvement. Nous allons donc utiliser directement nos composants géométriques pour construire notre modèle de région musculaire.

4.2 Modèle de force musculaire

Nous avons vu au chapitre 1 quelles étaient les caractéristiques d'un muscle : les attachements anatomiques, les fibres constituantes, les propriétés mécaniques et les propriétés électriques. Notre modèle étant basé sur des éléments géométriques, nous allons les utiliser directement pour modéliser les muscles.

La caractéristique fondamentale d'un muscle est que, lors de la contraction, la longueur des fibres diminue. Pour induire ce comportement contractile dans un tissu élastique, il suffit de

décrire un attracteur dans le sens des fibres du muscle modélisé. On utilise donc, ici encore, le principe de force engendrée par un attracteur.

Pour simplifier, dans notre modèle les directions de contractions sont définies localement pour chaque élément. On désigne les éléments voisins dans la direction desquels la contraction doit avoir lieu (voir Figure 13). Chaque élément voisin se trouvant dans la direction de contraction est défini comme attracteur.

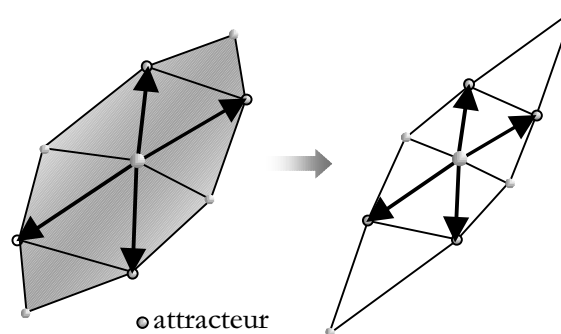


Figure 13. Modélisation de la force musculaire par des éléments attracteurs dans la direction de contraction des fibres.

En fait, il s'agit simplement de définir des directions de contraction à partir de la géométrie de l'objet et de positionner des ressorts entre les éléments correspondants. Notre modèle permet, si toutefois cela est nécessaire, d'utiliser les paramètres de forme (vus au paragraphe 3.2) pour définir précisément la direction des fibres sarcomères.

La pondération de la force musculaire permet de modéliser les phases d'activation, de contraction et de décontraction. Cette pondération varie au cours du temps et peut être différente pour chaque élément. Si la pondération est négative, on modélise une contraction active. Si la pondération est nulle, le retour à la configuration d'origine est garanti par la propriété d'élasticité de la région.

Contrairement à la force d'élasticité, la force musculaire permet de vérifier directement le principe d'action-réaction. Chaque force musculaire générée par un élément E_i sur un de ses voisins attracteur est automatiquement répercutée sur ce dernier (en inversant le signe du vecteur force).

Notons enfin qu'un muscle se déforme à volume constant, on pourra donc modéliser cette propriété si on le désire (voir chapitres 4 et 5).

Le principal inconvénient de notre méthode est que les paramètres des forces musculaires ne peuvent être déduits, comme dans (Chen, 1992), à partir des constantes classiques issues d'études physiologiques. Dans la méthode de (Chen, 1992), les forces provenaient d'équations du modèle de muscle de Zajac. Il est envisageable de reprendre cette étude afin de palier à l'inconvénient ci-dessus. Il faudrait définir, à partir des équations des forces musculaires, les pondérations de la force d'élasticité et de motricité.

L'avantage provient du fait que les forces motrices sont générées directement à partir de la géométrie. Nous verrons au chapitre suivant comment les transmettre facilement au squelette d'attachement.

4.3 Récapitulatif sur les éléments et régions musculaires

Un élément d'une région musculaire hérite de toutes les propriétés des éléments des régions élastiques. De plus il a ses propres caractéristiques qui sont : la liste des éléments voisins définissant les directions de contraction, et le poids courant de la force musculaire (pondération de la force d'attraction). Il est possible de privilégier des directions de contraction. Pour cela, il suffit de gérer les différentes pondérations associées à chacune des directions de contraction.

5. Régions solides

Une région solide est une région dans laquelle les distances entre les éléments qui la composent ne varient pas au cours de la simulation. Puisqu'on utilise l'approche par assemblage d'éléments, le plus simple est d'utiliser les algorithmes classiques de simulation d'objets solides (voir annexes 1 et 2).

À partir des éléments composant la région, on définit une orientation, une position de l'objet par rapport au repère global, la matrice d'inertie et les coordonnées des éléments dans le repère propre à la région (dont le centre est le barycentre et les axes les directions principales d'inertie). Lors de la simulation, l'ensemble des forces appliquées sur les éléments est pris en compte et la nouvelle orientation et la nouvelle position sont calculées.

La méthode utilisée ici est donc la méthode classique de simulation des objets rigides (voir annexes 1 et 2 pour les algorithmes de gestion des régions solides).

Dans le chapitre suivant, on expliquera comment on peut lier les régions solides aux régions élastiques ou aux autres régions solides.

5.1.1 Récapitulatif sur les éléments et régions solides

Les attributs complets des régions solides sont donnés à l'annexe 1. Les éléments des régions solides héritent des propriétés des éléments de type généraux. De plus, un autre attribut est calculé à l'initialisation : leur coordonnées dans le repère associé à la région.

Il faut noter, cependant, que la description surfacique sous forme de facettes triangulaires n'est pas obligatoire. La liste des voisins d'un élément solide peut ne contenir qu'un ou deux éléments, elle sert seulement pour l'affichage, puisque tous les calculs d'évolution se font par l'intermédiaire de l'isobarycentre de la région.

De même, on ne restreint pas la description des parties solides à la description de leurs surfaces, des éléments intérieurs peuvent être utilisés.

6. Conclusion

Dans ce chapitre, le modèle développé au cours de cette thèse a été présenté et comparé aux modèles existants. On a vu comment s'articulaient les différentes régions entre elles et ce qui les caractérisait (Figure 1). On a vu comment le principe de force défini par un point attracteur permettait la modélisation de l'élasticité et de la motricité (due à la contraction musculaire).

Cependant, le modèle en l'état n'est pas complet et ne suffit pas pour atteindre notre objectif.

Il nous manque :

- **Les liaisons entre les régions :** rien n'a encore été dit sur la manière de construire un objet à partir de plusieurs régions de propriétés distinctes.
- **L'incompressibilité :** cette propriété doit être appliquée sur une ou des régions élastiques définissant un même volume.

Nous verrons au chapitre suivant comment, par l'intermédiaire de contraintes, nous allons prendre en compte ces propriétés. Cela sera facilité par l'approche par assemblage d'éléments que nous avons suivie.

Nous verrons dans la dernière partie de ce manuscrit des exemples d'utilisation du modèle ainsi que le modèle du tronc développé à partir de celui-ci.

Chapitre 4

Contraindre le modèle

*Ce qui est simple est toujours faux,
Ce qui ne l'est pas est inutilisable.*

Paul Valery

Le chapitre précédent expliquait les bases du modèle que nous nous proposons de construire dans cette thèse. Nous avons comparé les parties originales du travail avec les méthodes rencontrées dans la littérature. Certaines propriétés sont modélisées intrinsèquement dans le modèle (l'élasticité, l'action musculaire ou la rigidité) en utilisant des régions de propriété homogène (voir figure 1, chapitre 3). Le raccordement de ces régions ainsi que certaines propriétés des matériaux que nous voulons modéliser n'ont pas encore été exposées (notamment la propriété d'incompressibilité). C'est l'objet de ce chapitre.

Ce chapitre est articulé de la manière suivante : nous montrons tout d'abord comment raccorder des régions de propriétés distinctes (paragraphe Chapitre 11), puis nous explicitons notre algorithme de résolution de contraintes (paragraphe Chapitre 12). Le paragraphe Chapitre 13 expose différents exemples de contraintes (locales et globales) et la façon de les résoudre. Pour finir, le paragraphe Chapitre 14 étudie plus en détail la contrainte d'incompressibilité.

1. Raccordement des régions

La simulation d'un objet complexe tel que le tronc passe par la simulation des différentes régions de cet objet. Au cours du précédent chapitre nous avons vu comment définir ces différentes régions de manière indépendante. Il faut maintenant étudier comment elles vont s'interfacer entre-elles, c'est-à-dire comment se raccordent des régions de propriétés différentes.

Trois types de raccordements sont à étudier : les raccordements entre deux régions élastiques (paragraphe 1.1), entre une région élastique et une région solide (paragraphe 1.2), et enfin entre deux régions solides (paragraphe 1.3).

1.1 Raccordement entre deux régions élastiques

Les régions musculaires sont aussi des régions élastiques. Elles héritent des propriétés des régions élastiques. On remarque que de par notre approche orientée objet, les régions musculaires sont traitées comme des régions élastiques.

Le raccordement traité dans ce paragraphe permet donc aussi de raccorder des régions élastiques et des régions musculaires.

Les régions élastiques décrivent les surfaces externes des objets qu'elles représentent. Le raccordement entre deux régions élastiques représentant deux objets en contact se fait de manière simple en continuité de la surface de contact. Les éléments à la frontière de deux régions, et appartenant à l'une des deux régions exclusivement, ont des voisins qui appartiennent à l'autre région. Ainsi, les éléments élastiques d'une région R_1 peuvent avoir des voisins dans une région élastique R_2 sans que l'algorithme d'évolution ne soit modifié (voir Figure 1).

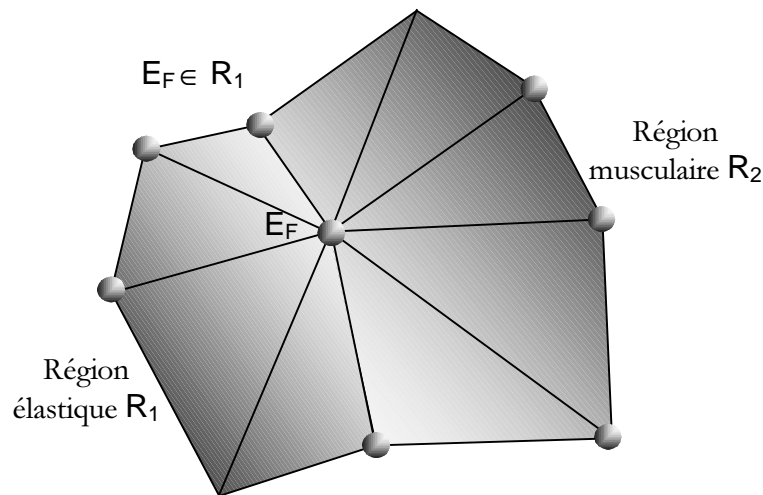


Figure 1. Raccordement de deux régions élastiques R_1 et R_2 . L'élément E_F est à la frontière entre R_1 et R_2 , $E_F \in R_1$ mais a des voisins dans R_2 .

1.2 Raccordement entre une région élastique et une région solide

Dans ce cas, le problème posé est le suivant : il faut maintenir le lien pendant la simulation quels que soient les mouvements des régions liées. Comme il est difficile de contraindre les mouvements des parties rigides, la méthode pour maintenir le lien durant la simulation consiste à contraindre les mouvements des éléments frontières de la région élastique. La méthode employée, voir (Promayon et al, 1997), est présentée au paragraphe 1.2.1, l'algorithme qui en découle au paragraphe 1.2.2.

1.2.1 Méthode

Soit R_e une région élastique et R_s une région solide que l'on veut maintenir accrochées (par exemple R_e représente un muscle attaché à un os du squelette représenté par R_s). La région solide R_s est définie par sa position et son orientation (voir chapitre 2 et annexe 1). Aux endroits de raccordement, les éléments sont doublés. C'est à dire qu'à chaque élément frontière E_i de R_e correspond un élément lié F_i de R_s . E_i et F_i ont toujours la même position. À chaque itération, F_i reçoit directement les forces exercées sur E_i . R_s prend en compte les forces exercées sur F_i et calcule sa nouvelle position et sa nouvelle orientation (F_i est déplacé). E_i est ensuite contraint à rester à la même position que F_i (on déplace E_i sur F_i).

1.2.2 Algorithme

L'algorithme de résolution du raccordement est le suivant (voir aussi illustration Figure 2) :

- Calculer séparément la somme des forces exercées sur chaque élément de R_e et de R_s (Figure 2.d),
- Calculer le mouvement de R_s (Figure 2.e),
- Calculer les mouvements des éléments de R_e (Figure 2.e),
- Pour chaque élément frontière E_i de R_e :
 - Contraindre la position de E_i à être identique à la position de l'élément F_i lié (déplacement de E_i en F_i) (Figure 2.f).

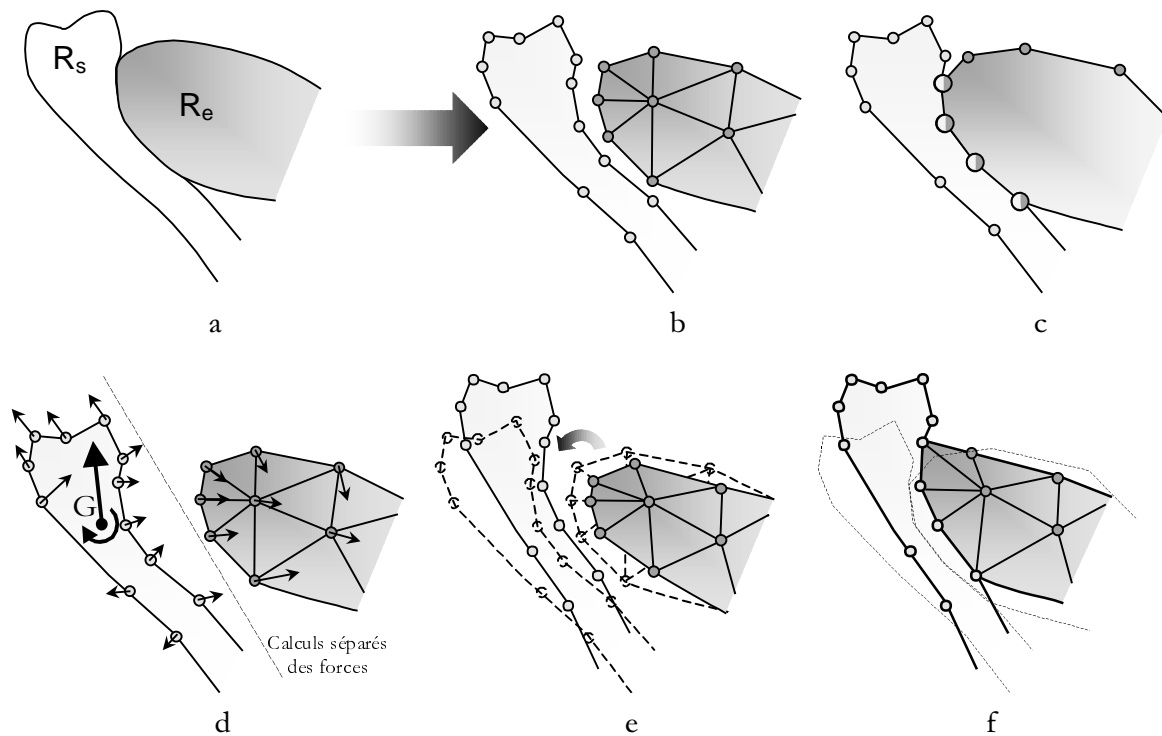


Figure 2. Raccordement entre une région élastique R_e et une région solide R_s . (a) géométrie initiale, (b) modélisation, (c) contrainte de raccordement (les éléments à la frontière de R_e sont forcés sur les position les éléments à la frontière de R_s), (d) (e) et (f) représentent une étape d'évolution (cf. algorithme 1.2.2).

Ainsi la force exercée sur les éléments E_i est directement transmise à la région R_s . Si une région musculaire (un muscle) est attachée sur une région solide (un os), la contraction musculaire entraînera, comme dans la réalité, un déplacement de la région solide. On peut aussi choisir de ne transmettre qu'une partie de la force de E_i (par exemple, pour modéliser des pertes d'énergie dans le raccordement).

1.3 Raccordement entre deux régions solides

Ce cas est celui de la modélisation d'objets rigides articulés (par exemple des chaînes composées de plusieurs maillons). On trouve de nombreuses solutions pour résoudre ce problème qui reviennent à déplacer les parties rigides tout en contraignant les liens : (Isaacs et Cohen, 1987), (Barzel et Barr, 1988), (van Overveld, 1991) et (Gascuel et Gascuel, 1994) (pour l'analyse de ces méthodes de contrainte, voir le chapitre 2). Toutes ces méthodes sont utilisables pour simuler des solides articulés à l'aide de notre modèle.

Ici, les régions rigides constituent le squelette. Les os du squelette sont reliés entre eux par une partie déformable : le cartilage. Ainsi, le raccordement entre deux régions solides ne doit

pas se traduire par une résolution exacte d'une contrainte de liaison, mais par la présence entre deux régions solides d'un lien déformable. C'est pourquoi nous choisissons de ne pas utiliser une contrainte point à point pour relier deux parties rigides. On utilise soit une région élastique dans le cas où la forme du cartilage est importante, soit un simple ressort entre deux éléments de deux régions solides à relier (voir Figure 3).

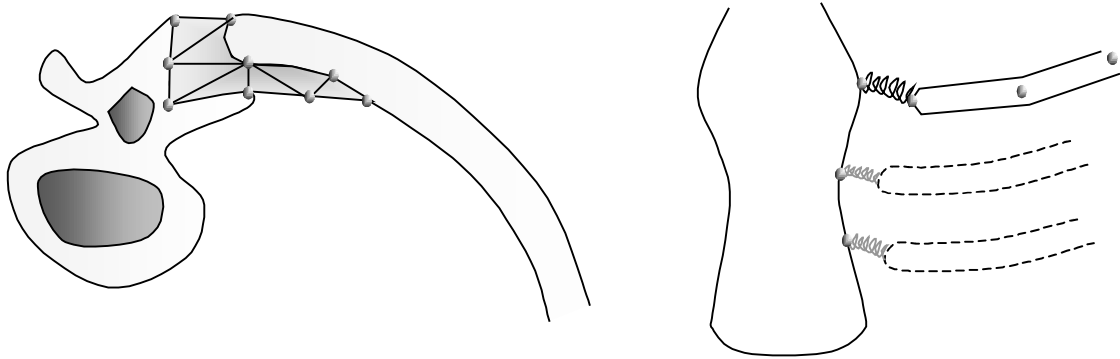


Figure 3. Raccordement entre deux régions solides (utilisation d'une région élastique à gauche ou d'un simple ressort à droite).

1.4 Conclusion

Que ce soit dans le cas d'une liaison entre une région élastique et une région solide ou d'une liaison par un ressort de deux régions solides entre-elles, les éléments frontières ou les éléments raccords ont besoin de connaître avec quel élément ils sont interfacés. Un nouvel attribut est défini pour chaque élément. Cet attribut désigne l'index d'un autre élément de l'objet avec lequel la liaison s'opère (voir annexe 4 pour un exemple de raccordement).

Nous avons examiné tous les cas possibles de raccordement entre les régions. Un seul des trois cas est traité comme une contrainte : le raccordement entre une région élastique et une région musculaire. Les éléments frontières des régions élastiques sont contraints sur leurs homologues des régions solides grâce à un déplacement. La manière dont nous avons résolu cette contrainte donne un aperçu succinct de la méthode de résolution de contrainte privilégiée dans cette thèse. Nous allons maintenant décrire cette méthode avec précision, permettant, dans un cadre plus général, de résoudre les autres contraintes nécessaires à notre modélisation.

2. Résolution des contraintes

Nous avons examiné au chapitre 2 les différentes solutions pour contraindre des modèles générateurs évoluant à l'aide de lois basées sur la physique. Nous avons choisi dans cette thèse d'opter pour la méthode de résolution directe car elle permet une résolution exacte de la contrainte sans utiliser un algorithme itératif. La suite de ce chapitre s'attachera à montrer les nombreuses possibilités offertes par cette méthode, aussi bien pour résoudre des contraintes locales que des contraintes globales. Ce paragraphe s'appuie sur le paragraphe concernant les contraintes du chapitre 2 (paragraphe 3).

Nous expliquons tout d'abord le principe général issu de la formulation dynamique de la résolution de contrainte (paragraphe 2.1). Puis, nous expliquons le cadre général de la méthode de résolution directe au paragraphe 2.2. Nous utilisons un exemple simple de minimisation sous contrainte afin d'analyser notre méthode par rapport à deux autres méthodes issues de la littérature (paragraphe 2.3). Au paragraphe 2.4, nous discutons les possibilités pour résoudre plusieurs contraintes à l'aide de cette technique.

2.1 Formulation dynamique de la résolution de contrainte

Au chapitre 2, on a vu différentes méthodes de résolution de contraintes. Nous allons examiner le mécanisme d'application d'une contrainte différentiable une fois et énoncer une propriété importante (voir aussi (Witkin et al, 1995)).

Soit un objet de masse m et de vecteur position \mathbf{x} soumis à un ensemble de forces \mathbf{f} . On discrétise la loi d'évolution de l'objet pour obtenir ses positions successives \mathbf{x}^t au cours du temps.

On désire que cet objet vérifie une contrainte de manière forte à chaque pas de la discrétisation. C'est-à-dire, on désire qu'à tout instant de la simulation une contrainte C soit vérifiée. Prenons le cas d'une contrainte exprimée sur la position de l'objet (par exemple l'objet doit toujours se trouver dans une région de l'espace).

On exprime la contrainte implicitement sous forme mathématique : $C(\mathbf{x}) = 0$. Examinons les conséquences de cette contrainte. On désire formuler la force \mathbf{f}_C due à la contrainte à chaque instant de la simulation. On sépare les forces dues aux contraintes \mathbf{f}_C des autres forces connues \mathbf{f}_E (forces engendrées par l'élasticité, la gravité...) dans la relation fondamentale de la dynamique. Cette dernière s'exprime alors :

$$m\ddot{\mathbf{x}} = \mathbf{f}_E + \mathbf{f}_C.$$

On désire que la contrainte soit toujours vérifiée :

$$\forall t, C(\mathbf{x}^t) = 0 \Leftrightarrow \begin{cases} C(\mathbf{x}^{t_0}) = 0 \\ \dot{C}(\mathbf{x}^t) = 0 \end{cases}.$$

Considérons l'énergie cinétique de l'objet :

$$E_{cin} = \frac{1}{2}m\dot{\mathbf{x}}^2.$$

On peut exprimer la variation d'énergie cinétique du système par :

$$\frac{\partial E_{cin}}{\partial t} = \langle m\ddot{\mathbf{x}}, \dot{\mathbf{x}} \rangle = \langle \mathbf{f}_E + \mathbf{f}_C, \dot{\mathbf{x}} \rangle = \langle \mathbf{f}_E, \dot{\mathbf{x}} \rangle + \langle \mathbf{f}_C, \dot{\mathbf{x}} \rangle = \langle \mathbf{f}_E, \dot{\mathbf{x}} \rangle + \frac{\partial E_{cin}^C}{\partial t}.$$

On désire que les forces dues aux contraintes et calculées par la méthode de résolution n'influent *jamais* sur l'énergie cinétique du système, c'est-à-dire que :

$$\forall t, \frac{\partial E_{cin}^C}{\partial t} = \langle \mathbf{f}_C^t, \dot{\mathbf{x}}^t \rangle = 0.$$

Or comme on veut que la contrainte soit *toujours* vérifiée, c'est-à-dire que :

$$\forall t, C(\dot{\mathbf{x}}) = 0 \Rightarrow \frac{\partial C}{\partial t} = 0 \Rightarrow \left\langle \frac{\partial C}{\partial \mathbf{x}}, \frac{\partial \mathbf{x}}{\partial t} \right\rangle = 0.$$

Ce qui nous donne la condition suivante :

$$\forall t, \langle \mathbf{f}_c^t, \dot{\mathbf{x}} \rangle = 0 \text{ tel que } \left\langle \frac{\partial C}{\partial \mathbf{x}}, \dot{\mathbf{x}} \right\rangle = 0.$$

Pour que cette condition soit vérifiée, on doit avoir :

$$\mathbf{f}_c - \lambda \frac{\partial C}{\partial \mathbf{x}} = 0.$$

C'est-à-dire que *les forces dues à la contrainte sont colinéaires au gradient de l'expression de la contrainte.*

Pour résoudre une contrainte on doit donc calculer, pour un \mathbf{x}^t donné, la valeur du coefficient λ . Cette démonstration peut être faite de manière plus générale lorsque plusieurs contraintes sont appliquées sur l'objet, voir (Witkin et al, 1995).

2.2 Principe de la résolution directe

Nous avons vu au chapitre 2 (paragraphe 3) qu'il y avait trois approches principales pour résoudre des contraintes :

- **Méthodes d'optimisation** : lorsque la simulation physique est pilotée par une énergie (énergie de déformation, de mouvement et énergie potentielle), l'optimisation (minimisation) de la fonction d'énergie sous contraintes permet de décrire les différents états des objets. Les méthodes d'optimisation sous contraintes sont nombreuses dans la littérature mathématique. Pour la plupart, elles utilisent les multiplicateurs de Lagrange. (Platt et Barr, 1988) ont développé une de ces méthodes pour les modèles déformables. (Fua et Bechbühler, 1997) ont eux aussi mis au point une nouvelle technique qu'ils ont utilisée en vision pour la segmentation d'image par contour actif. Ces deux méthodes seront comparées avec la méthode de résolution directe au paragraphe 2.3.
- **Forces de contraintes** : ici les forces engendrées par les contraintes sont calculées (par la dynamique inverse ou une approche itérative) puis appliquées aux objets.
- **Déplacements dus aux contraintes** : le principe est le même que le précédent, mais au lieu de calculer les forces à appliquer pour vérifier les contraintes, ce sont les déplacements qui sont calculés. La méthode de résolution directe entre dans cette catégorie.

Dans ce paragraphe nous dégagons d'abord l'algorithme de résolution utilisé dans la résolution directe (paragraphe 2.2.1). Cet algorithme calcule directement les déplacements dus aux contraintes par une méthode de projection. La direction de projection est expliquée au paragraphe 2.2.2. Nous explicitons ensuite le système d'équations résultant (paragraphe 2.2.3). Nous concluons en dégageant un bilan sur les avantages et les inconvénients (paragraphe 2.2.4 et 2.2.5) de cette approche, puis nous proposons une solution pour la détection des problèmes de discontinuités (paragraphe 2.2.6).

2.2.1 Algorithme

Dans une approche par assemblage d'élément utilisant un schéma eulérien, l'algorithme de simulation classique à chaque nouvelle itération est le suivant : pour chaque élément calculer

les forces qui leur sont appliquées, et discrétiser ces forces pour calculer leurs nouvelles positions (voir chapitre 2, paragraphe 2.2).

Lorsque le système est contraint, l'algorithme doit prendre en compte les contraintes. Il est quelque peu différent (voir illustration Figure 4).

À un instant t de la simulation :

- Pour chaque élément, calculer les forces \mathbf{F}^t qui leur sont appliquées.
- Utiliser un schéma de discrétisation (type Euler) pour calculer un état cinétique intermédiaire $\hat{\mathbf{P}}$ (cet état peut violer les contraintes).
- Calculer, à partir de l'état intermédiaire, les déplacements éventuelles δ_C nécessaires à la vérification des contraintes.
- Déplacer les éléments selon δ_C : l'état cinétique final est calculé.
- Dédire la vitesse réelle à partir de la nouvelle position.

Nous verrons dans les deux paragraphes suivant les détails des calculs.

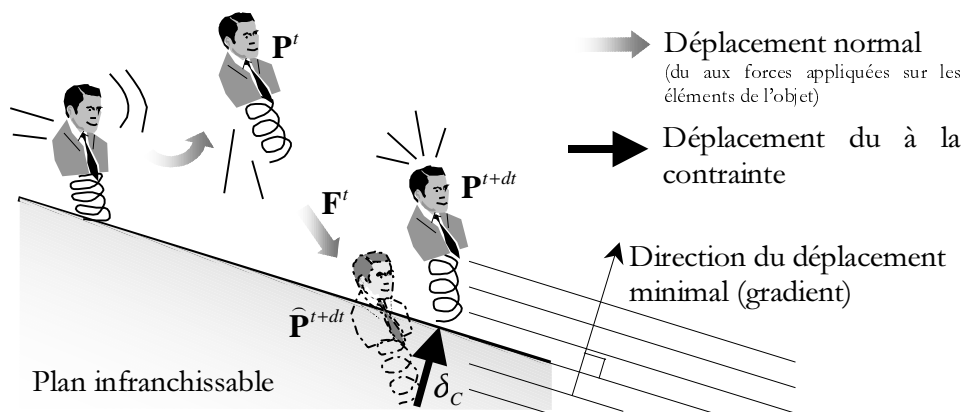


Figure 4. Exemple de résolution de contrainte par la méthode directe. La contrainte est un plan incliné impénétrable. Au cours de la simulation, l'objet franchi le plan et viole la contrainte. Pour résoudre la contrainte l'objet est déplacé (de manière minimale) afin de vérifier à nouveau la contrainte.

2.2.2 Choix de la direction

À l'étape 3 de l'algorithme on doit trouver la direction de déplacement qui vérifie les contraintes.

Soit un objet décrit par son vecteur position (de dimension $3n$ si le nombre d'éléments le composant est n), soit \mathbf{x}^t la position au temps t , $\hat{\mathbf{x}}^{t+dt}$ la position intermédiaire calculée sans

tenir compte des contraintes et $C(\mathbf{x}^t)$ l'expression de la contrainte (la contrainte est vérifiée lorsque $C(\mathbf{x}^t) = 0$). Le problème se décrit par le système suivant :

$$\begin{cases} \mathbf{x}^{t+dt} = \widehat{\mathbf{x}}^{t+dt} + \delta_c \\ C(\mathbf{x}^{t+dt}) = 0 \end{cases}$$

La première équation indique que la nouvelle position est la position intermédiaire plus un déplacement dû à la vérification de la contrainte δ_c . La deuxième équation exprime que la nouvelle position doit vérifier la contrainte C .

L'élément essentiel de la résolution directe revient donc à trouver un vecteur δ_c satisfaisant. Les critères définissant ce vecteur sont les suivants :

- **Minimiser les conséquences des déplacements :** l'énergie apportée par le déplacement dû aux contraintes doit être minimal. On désire que l'influence de la résolution des contraintes sur l'état cinétique de l'objet soit la plus petite possible pour éviter un trop gros saut d'énergie dans l'état de l'objet. En effet, dans les systèmes définis par des modèles générateurs, les sauts d'énergie (dus à un choc ou un apport brusque d'une nouvelle force) sont déstabilisateurs.
- **Vérifier la contrainte de manière exacte :** on cherche une solution exacte.
- **Trouver le déplacement en une passe :** pour augmenter la rapidité de résolution, on doit trouver *directement* le déplacement total (sans processus itératif).

On a choisi de prendre un vecteur δ_c colinéaire à la direction du gradient de C en $\widehat{\mathbf{x}}^{t+dt}$ (quand cela est possible, en \mathbf{x}^t sinon). Deux arguments sont en faveur de ce choix :

- On a vu, au paragraphe 2.1, que les forces dues aux contraintes doivent être colinéaires à la direction du gradient de la contrainte pour que leur travail soit nul. Ainsi ces forces n'apportent pas d'énergie supplémentaire au système. Dans notre méthode, le choix est fait de ne pas calculer ces forces dues aux contraintes, mais de calculer directement les déplacements dus aux contraintes (par l'intermédiaire de ces forces). Comme on utilise une formulation eulérienne de la dynamique (discrétisation temporelle de la loi fondamentale de la dynamique), le déplacement déduit des forces sera dans la même direction que celles-ci : la direction du gradient de la contrainte.
- La direction du gradient en $\widehat{\mathbf{x}}^{t+dt}$ est mathématiquement la direction de déplacement minimum la plus simple, compte tenu de son efficacité, que l'on puisse trouver en une passe. Cela reste vrai tant que le champ de gradient est continu (voir Figure 6).

2.2.3 Résolution

Ainsi, en posant $\delta_c = \lambda \cdot \nabla C(\widehat{\mathbf{x}}^{t+dt})$, le problème se reformule de la manière suivante :

$$\begin{cases} \mathbf{x}^{t+dt} = \widehat{\mathbf{x}}^{t+dt} + \lambda \cdot \nabla C(\widehat{\mathbf{x}}^{t+dt}) \\ C(\mathbf{x}^{t+dt}) = 0 \end{cases},$$

où $\nabla C(\mathbf{x})$ est le gradient de la fonction de contrainte en \mathbf{x} , c'est-à-dire :

$$\nabla C(\mathbf{x}) = \frac{\partial C}{\partial \mathbf{x}},$$

et λ le coefficient multiplicateur du gradient. Ce système est un système de $3n+1$ équations à $3n+1$ inconnues (\mathbf{x}^{t+dt} et λ). On utilise soit des méthodes classiques de résolution, soit on explicite d'abord la valeur de λ (notamment quand l'expression de la contrainte a un degré inférieur ou égal à 4). Pour cela on cherche la solution du polynôme suivant :

$$C(\widehat{\mathbf{x}}^{t+dt} + \lambda \cdot \nabla C(\widehat{\mathbf{x}}^{t+dt})) = 0$$

Cette dernière technique permet de trouver λ de manière directe et donc de calculer en une seule passe les nouvelles coordonnées vérifiant les contraintes. On peut interpréter cette résolution en remarquant que \mathbf{x}^{t+dt} est l'intersection entre la droite de direction $\nabla C(\widehat{\mathbf{x}}^{t+dt})$ et le sous-espace défini par la contrainte $C(\mathbf{x})=0$ (voir Figure 5). Plus le degré de la fonction de contrainte est simple, plus l'intersection est facile à calculer de manière directe.

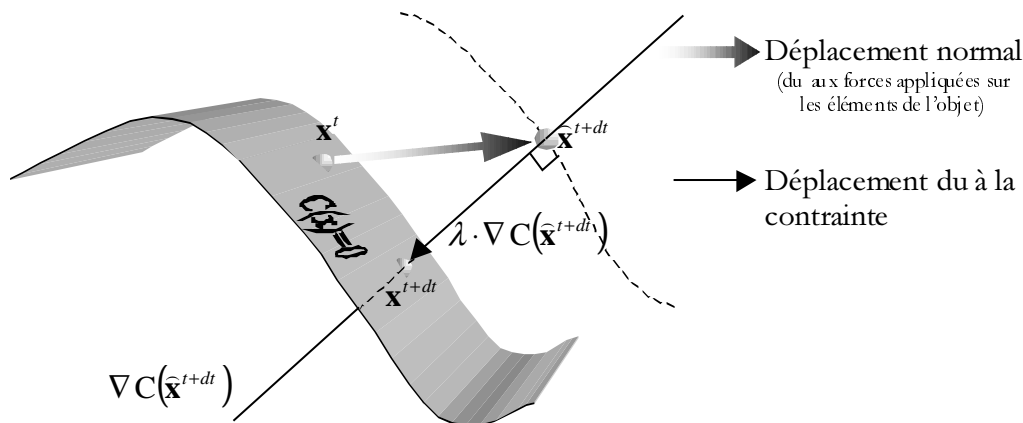


Figure 5. Méthode directe par projection sur la contrainte. On cherche l'intersection entre la droite de vecteur directeur $\nabla C(\widehat{\mathbf{x}}^{t+dt})$ passant par $\widehat{\mathbf{x}}^{t+dt}$ et la courbe de contrainte définie par le sous-espace $C(\mathbf{x})=0$.

2.2.4 Avantages

Alors que dans les méthodes lagrangiennes (utilisation d'un multiplicateur de Lagrange et d'un algorithme de minimisation) une solution approximative est trouvée après un certain nombre d'itérations, ici la contrainte est exactement vérifiée en une seule passe. La méthode est donc plus précise et plus rapide.

Dans le cas où la région n'est qu'un point \mathbf{x}_c de l'espace des paramètres, on choisit

$$\delta_c = \mathbf{x}_c - \widehat{\mathbf{x}}^{t+dt}.$$

Remarquons aussi que la contrainte peut être exprimée sur un autre paramètre que la position.

2.2.5 Inconvénients

On examine deux types d'inconvénients différents dans ce paragraphe.

D'une part, les contraintes doivent vérifier des propriétés particulières pour pouvoir être utilisé dans notre algorithme. Les propriétés suivantes sont indispensables :

- Contrainte différentiable au moins une fois. On doit exprimer la contrainte comme une fonction du paramètre sur lequel elle s'applique et pouvoir trouver le jacobien de la contrainte par rapport au paramètre. Dans le cas où cela n'est pas possible, il faut utiliser un algorithme itératif d'ajustement de la position, voir (Gascuel et Gascuel, 1994).
- Contrainte continue. La contrainte doit être continue de telle sorte que le gradient soit toujours utilisable quel que soit $\widehat{\mathbf{x}}^{t+dt}$. Un contre exemple est présenté à la Figure 6. Dans ce cas, pour résoudre la contrainte, on doit soit mettre en place une détection et un traitement des cas singuliers (si $\widehat{\mathbf{x}}^{t+dt}$ est dans la zone de discontinuité), soit utiliser un algorithme d'optimisation itératif (dans notre cas un algorithme de descente de gradient).

Le paragraphe suivant tente de palier à ce dernier inconvénient en proposant une détection des zones de discontinuité.

D'autre part, la *réponse dynamique* due à la contrainte n'est pas calculée. C'est un inconvénient par rapport à l'aspect physique du modèle : la validité physique de l'évolution n'est plus complètement assumée.

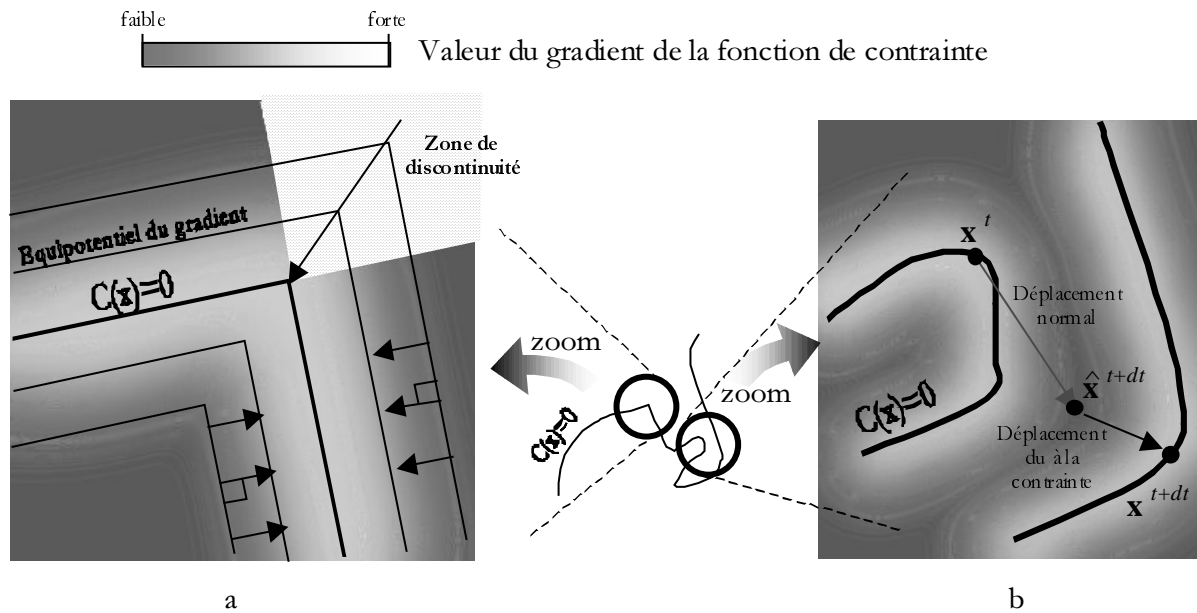


Figure 6. Problèmes induits par une discontinuité de la fonction de contrainte et de son gradient (à gauche) ou par une forme particulière de la contrainte (à droite) qui occasionne un trop grand changement d'état.

2.2.6 Contrôle supplémentaire et détection des zones de discontinuité

La Figure 6 présente deux cas où la forme de la fonction de contrainte peut induire des changements d'état indésirable. Une courbe de contrainte est représentée au centre de la figure. Deux endroits particuliers sont grossis de part et d'autres de cette courbe. À gauche, Figure 6a, est présenté le cas d'une zone de discontinuité dans le gradient de la courbe de contrainte. À droite, Figure 6b, est présenté un cas où la projection sur la courbe de contrainte induit un changement d'état important. Ce changement d'état important (du essentiellement à la résolution de la contrainte par projection) implique une variation importante de l'objet (et de sa géométrie), voire une discontinuité.

Pour éviter d'éventuels changements d'état trop importants (donc des déplacements trop grands), on ajoute une contrainte dite de « cohérence temporelle ». La contrainte de cohérence temporelle est vérifiée lorsque :

$$\|\mathbf{x}^t - \hat{\mathbf{x}}^{t+dt}\| \geq \|\mathbf{x}^t - \mathbf{x}^{t+dt}\|.$$

C'est-à-dire que le déplacement dû aux forces seules (avant résolution de contrainte) est plus grand que le déplacement final. Ainsi, la résolution de la contrainte ne déplace pas l'objet plus loin qu'un déplacement non contraint. Dans le cas où la cohérence temporelle n'est pas

vérifiée (voir Figure 6b), le pas de temps est diminué et on réitère l'algorithme présenté au paragraphe 2.2.1 .

Notons que si une valeur limite dt_{\min} du pas de temps est atteinte sans que la cohérence temporelle ne soit vérifiée, on se trouve dans un cas singulier (une zone de discontinuité du gradient, Figure 6a). Cela permet de détecter les éventuelles zones de discontinuité.

Dans les contraintes classiques (comme celle de déformation à volume constant), ce cas ne se présente jamais (la fonction de contrainte est continue). Par exemple, dans le cas de la contrainte d'incompressibilité (comme nous le verrons au paragraphe 4), le système à résoudre (présenté au paragraphe 2.2.3) revient à trouver la plus petite racine d'un polynôme de degré trois en λ . Cette contrainte est donc continue, et les cas de la Figure 6 ne sont jamais rencontrés.

2.3 Comparaison

Nous allons maintenant comparer la méthode directe avec deux autres méthodes issues de la littérature : (Platt et Barr, 1988) et (Fua et Bechbühler, 1997). On notera que la méthode directe est similaire aux méthodes de calcul par la dynamique inverse des forces dues aux contraintes, comme celle présentée par (Barzel et Barr, 1988), voir chapitre 2.

Comme les deux articles choisis pour la comparaison proposent des algorithmes de minimisation sous contrainte, on va considérer un problème du type :

$$\text{Minimiser } f(\mathbf{x}) \text{ sous la contrainte } g(\mathbf{x})=0,$$

À l'aide d'un exemple tiré de (Fua et Bechbühler, 1997).

La méthode de résolution directe développée ici ne répond pas aux problèmes de minimisation. On l'utilise dans un tel problème pour la comparer aux autres méthodes. Cependant, le but de ce paragraphe n'est pas de déterminer les critères de choix pour l'application de l'une ou l'autre des méthodes. L'objectif de la comparaison est plus explicatif. Ce paragraphe doit permettre de mettre en lumière le fonctionnement de la méthode de résolution directe par rapport aux autres méthodes.

Ce paragraphe est organisé de la manière suivante : nous décrivons d'abord le problème choisi (paragraphe 2.3.1) avant d'examiner le comportement de chacune des trois méthodes séparément (paragraphe 2.3.2 , 2.3.3 , et 2.3.4) puis nous comparerons les résultats (paragraphe 2.3.5).

2.3.1 Problème

On reprend un exemple simple de minimisation d'une fonction, tiré de (Fua et Bechbühler, 1997). On se place dans le plan. Il s'agit de minimiser la fonction :

$$f(\mathbf{x}) = (x - 0,5)^2 + (y - 0,2)^2, \text{ sous la contrainte :}$$

$$g(\mathbf{x}) = \frac{1}{4}x^2 + y^2 - 1 = 0. \text{ (voir Figure 7)}$$

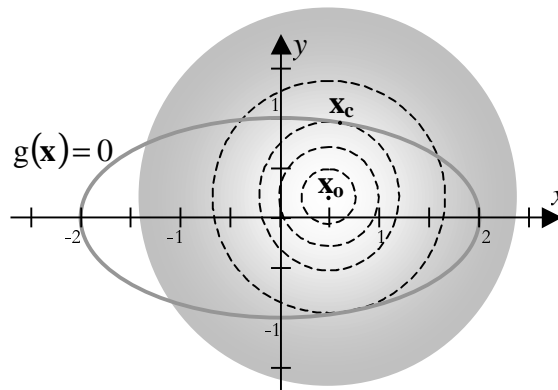


Figure 7. Fonction à minimiser sous contrainte. L'optimum de $f(\mathbf{x})$ est le point $\mathbf{x}_0 = (0,5 \ 0,2)$. L'optimum de $f(\mathbf{x})$ vérifiant $g(\mathbf{x})$ est \mathbf{x}_c .

Par la suite nous aurons besoin de :

$$\nabla f(\mathbf{x}) = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} = \begin{pmatrix} \frac{\partial f(\mathbf{x})}{\partial x} \\ \frac{\partial f(\mathbf{x})}{\partial y} \end{pmatrix} = \begin{pmatrix} 2x - 1 \\ 2y - \frac{2}{5} \end{pmatrix} \text{ et de } \nabla g(\mathbf{x}) = \begin{pmatrix} \frac{1}{2}x \\ 2y \end{pmatrix}.$$

On utilisera toujours le même point initial $\mathbf{x}_i = (-1,6 \ 0,45)$. Tous les algorithmes présentés utilisent un pas k (c'est le paramètre de l'algorithme), nous avons choisi dans un premier temps $k = 0,45$ pour tous les algorithmes.

Pour tous les algorithmes on présente l'évolution des fonctions $f(\mathbf{x})$ et $g(\mathbf{x})$ et de \mathbf{x} au cours des itérations des algorithmes de résolution.

2.3.2 Multiplicateur de Lagrange (Platt et Barr, 1988)

Cette méthode consiste à considérer la solution du problème de minimisation sous contrainte comme un point critique de la fonction définie par :

$$\mathcal{E}_{lagrange}(\mathbf{x}) = f(\mathbf{x}) + \lambda \cdot g(\mathbf{x}).$$

λ est appelé multiplicateur de Lagrange pour la contrainte $g(\mathbf{x}) = 0$.

À partir de l'équation énergétique ci-dessus, pour \mathbf{x} extrémal, on a :

$$\nabla \mathcal{E}_{lagrange}(\mathbf{x}) = 0 = \nabla f(\mathbf{x}) + \lambda \cdot \nabla g(\mathbf{x}).$$

C'est-à-dire que le gradient de $f(\mathbf{x})$ est colinéaire au gradient de $g(\mathbf{x})$ quand $\mathcal{E}_{lagrange}(\mathbf{x})$ est extrémal. La constante de proportionnalité entre $\nabla f(\mathbf{x})$ et $\nabla g(\mathbf{x})$ est $-\lambda$.

Les auteurs utilisent une alternative à la descente de gradient où λ remonte la pente du gradient. Le minimum de la contrainte devient alors un point d'attraction au lieu d'être un point de répulsion, ce qui permet bien de résoudre le problème. On aboutit à :

$$\begin{cases} \dot{\mathbf{x}} = -\frac{\partial f}{\partial \mathbf{x}} - \lambda \frac{\partial g}{\partial \mathbf{x}} \\ \dot{\lambda} = +g(\mathbf{x}) \end{cases}$$

L'algorithme consiste à utiliser ces dérivées itérativement :

$$\begin{cases} x^{i+1} = x^i + k \cdot \left(-\left(2 + \frac{\lambda^i}{2} \right) x^i + 1 \right) \\ y^{i+1} = y^i + k \cdot \left(-2(1 + \lambda^i) y^i + \frac{2}{5} \right) \\ \lambda^{i+1} = \lambda^i + k \cdot \left(\left(\frac{x^i}{2} \right)^2 + y^2 - 1 \right) \end{cases}$$

On choisit $\lambda^0 = 0$.

Les résultats sont présentés à la Figure 8.

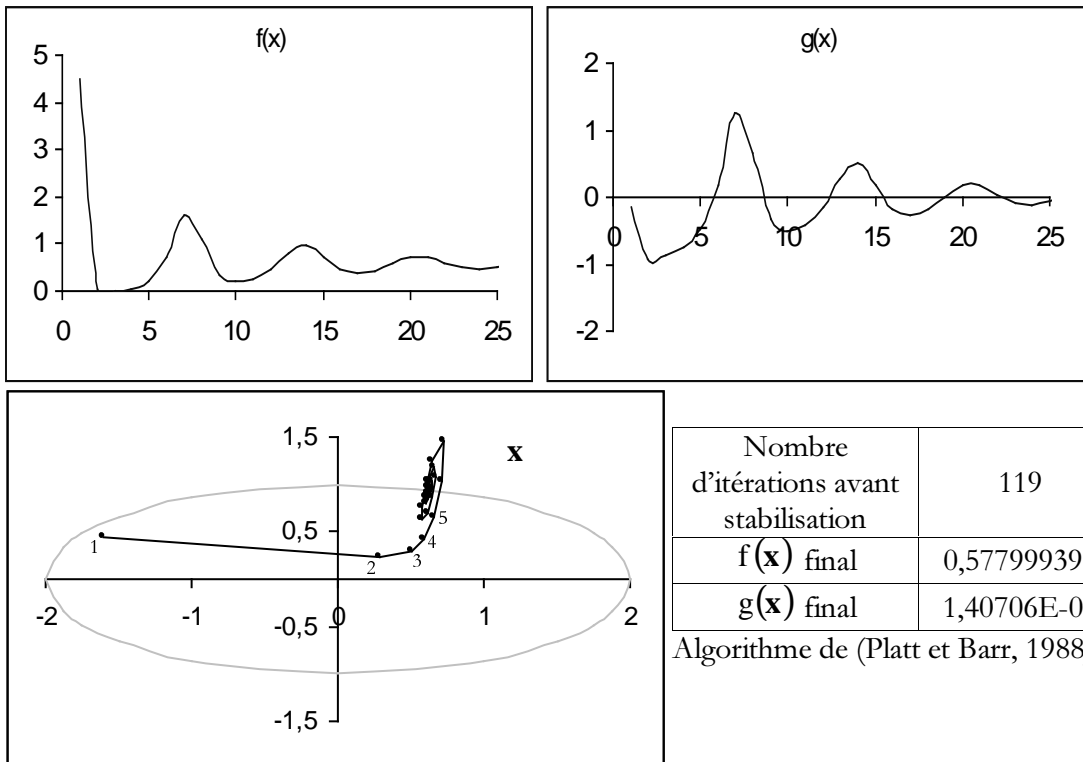


Figure 8. Variations $f(\mathbf{x})$, $g(\mathbf{x})$ et de \mathbf{x} au cours des itérations. Sur la courbe d'évolution de \mathbf{x} , les résultats des premières itérations sont annotés par le numéro d'itération correspondant.

2.3.3 Optimisation dans le sous-espace orthogonal (Fua et Bechbühler, 1997)

Cet algorithme a été utilisé pour la segmentation par contour actif (pour les contours actifs voir (Kass et al, 1988) et (Leitner, 1993)). Les contours actifs minimisent une énergie somme de l'énergie interne (due à la déformation) et de l'énergie externe (ou énergie potentielle contenue dans une image à segmenter). Comme ces méthodes sont sujettes à des problèmes de stabilité et sensibles au bruit contenu dans les images à segmenter, l'évolution des contours actifs doit être contrainte. En vision, les solutions de contraintes les plus utilisées sont les méthodes de pénalités. Pour tirer parti des propriétés spécifiques des contours actifs, P. Fua et C. Brechbühler ont mis au point un algorithme de minimisation sous contrainte particulier.

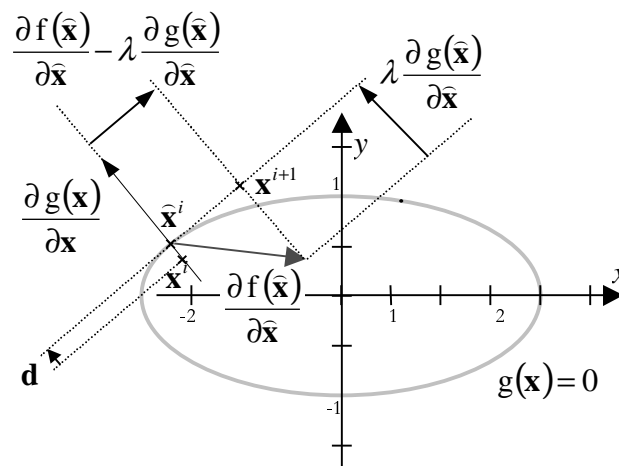


Figure 9. Géométrie de la méthode d'optimisation dans le sous-espace orthogonal.

Comme dans notre approche, l'algorithme se déroule en deux étapes :

- Forcer l'objet sur la contrainte, en projetant l'état courant sur la surface de contrainte. On estime le vecteur de projection \mathbf{d} en partant de :

$$g(\mathbf{x} + \mathbf{d}) = g(\mathbf{x}) + \frac{\partial g(\mathbf{x})}{\partial \mathbf{x}} \cdot \mathbf{d} \quad (\text{approximation de la série de Taylor}),$$

or comme on veut $g(\mathbf{x} + \mathbf{d}) = 0$, on a :

$$\frac{\partial g(\mathbf{x})}{\partial \mathbf{x}} \cdot \mathbf{d} = -g(\mathbf{x}).$$

Ce qui permet de trouver \mathbf{d} et donc une position intermédiaire $\hat{\mathbf{x}} = \mathbf{x} + \mathbf{d}$.

- Minimiser la fonction $f(\mathbf{x})$ en projetant le gradient de cette fonction sur le sous-espace tangent à la surface de la contrainte et en cherchant la direction de projection de telle façon que le résultat final ne soit pas trop éloigné de la surface de contrainte. Pour cela, on va projeter $\nabla f(\mathbf{x})$ sur la droite tangente à $g(\mathbf{x})$ au point $\hat{\mathbf{x}}$ (voir Figure 9) en cherchant à résoudre le système sur-déterminé :

$$\frac{\partial f(\hat{\mathbf{x}})}{\partial \hat{\mathbf{x}}} = \lambda(\hat{\mathbf{x}}) \frac{\partial g(\hat{\mathbf{x}})}{\partial \hat{\mathbf{x}}},$$

au sens des moindres carrés. Cela se fait en résolvant :

$$\left\langle \frac{\partial f(\hat{\mathbf{x}})}{\partial \hat{\mathbf{x}}}, \frac{\partial g(\hat{\mathbf{x}})}{\partial \hat{\mathbf{x}}} \right\rangle = \lambda \cdot \left\langle \frac{\partial g(\hat{\mathbf{x}})}{\partial \hat{\mathbf{x}}}, \frac{\partial g(\hat{\mathbf{x}})}{\partial \hat{\mathbf{x}}} \right\rangle.$$

Puis on calcule le nouvel état en posant : $\mathbf{x}^{i+1} = \hat{\mathbf{x}}^i + k \cdot \left(\frac{\partial f(\hat{\mathbf{x}}^i)}{\partial \hat{\mathbf{x}}} - \lambda(\hat{\mathbf{x}}^i) \frac{\partial g(\hat{\mathbf{x}}^i)}{\partial \hat{\mathbf{x}}} \right)$.

Appliquons cette méthode à notre exemple :

Calcul de \mathbf{d} :

On choisit le vecteur \mathbf{d} le plus petit possible, pour cela il faut qu'il soit colinéaire au gradient de la contrainte. C'est-à-dire qu'il faut que :

$$\mathbf{d} = \frac{\partial g(\mathbf{x})}{\partial \mathbf{x}} \cdot \Delta_x,$$

où Δ_x est un scalaire. On a donc :

$$\frac{\partial g(\mathbf{x})}{\partial \mathbf{x}} \cdot \mathbf{d} = \frac{\partial g(\mathbf{x})}{\partial \mathbf{x}} \cdot \frac{\partial g(\mathbf{x})}{\partial \mathbf{x}} \cdot \Delta_x = -g(\mathbf{x}).$$

Dans cet exemple, après résolution, on trouve :

$$\Delta_x = \frac{-\frac{x^2}{4} - y^2 + 1}{\frac{x^2}{4} + 4y^2} \text{ et on prend } \mathbf{d} = \begin{pmatrix} \frac{1}{2}x \\ 2y \end{pmatrix} \cdot \Delta_x.$$

Calcul de $\hat{\mathbf{x}}$

La première étape de l'algorithme se termine par le calcul de :

$$\hat{\mathbf{x}} = \mathbf{x} + \mathbf{d}.$$

Expression de λ

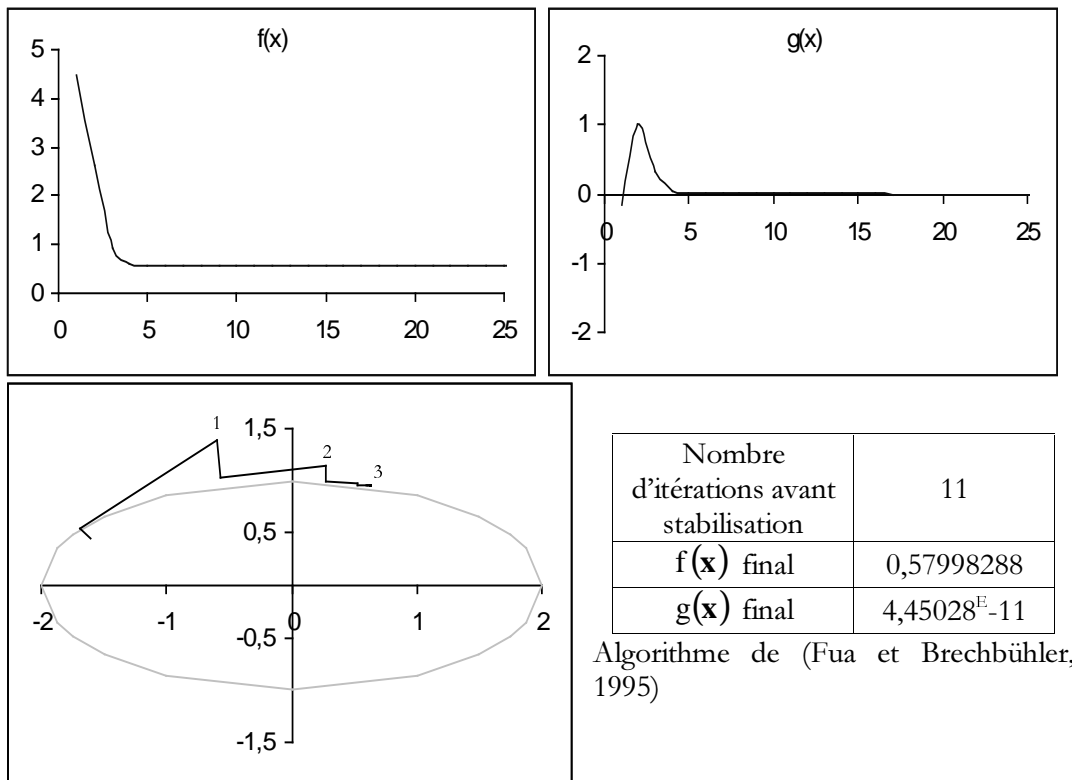
Après résolution, on trouve :

$$\lambda = \frac{\hat{x}^2 - \frac{1}{2}\hat{x} + 4\hat{y}^2 - \frac{4}{5}\hat{y}}{\frac{1}{4}\hat{x}^2 + 4\hat{y}^2}.$$

Calcul de la nouvelle position

$$\mathbf{x}^{i+1} = \hat{\mathbf{x}} + k \cdot \begin{pmatrix} \left(2 - \frac{\lambda}{2}\right)\hat{x} - 1 \\ 2(1 - \lambda)\hat{y} - \frac{2}{5} \end{pmatrix}$$

Les résultats de cet algorithme sont montrés à la Figure 10.



Algorithme de (Fua et Brechbühler, 1995)

Figure 10. Variations de $f(\mathbf{x})$, $g(\mathbf{x})$ et de \mathbf{x} au cours des itérations.

2.3.4 Méthode de résolution directe

La méthode présentée au paragraphe 2.2 n'a pas été vue dans le cadre d'une minimisation de fonction, mais dans celui de résolution de contrainte au cours d'un algorithme d'évolution. On utilise le principe de point d'attraction pour l'utiliser en tant que méthode d'optimisation et ainsi la comparer aux deux précédentes.

Le principe du point d'attraction nous indique la direction dans laquelle l'objet doit se déplacer (équivalent à l'ajout d'une force toujours dirigée vers le point \mathbf{x}_c). En fait on se dirige dans la direction opposée au gradient, comme dans la méthode de (Platt et Barr, 1988). Cependant l'algorithme utilise selon la méthode de résolution directe, il comporte deux étapes, comme (Fua et Bechbühler, 1997) :

- Déplacement dans la direction du point d'attraction, pour cela on utilise $-\nabla f(\mathbf{x})$, calcul d'un état intermédiaire ne vérifiant pas les contraintes,
- Projection sur la contrainte par résolution directe. Cela se fait en résolvant le système :

$$\begin{cases} g(\mathbf{x}^{i+1}) = 0 \\ \mathbf{x}^{i+1} = \hat{\mathbf{x}} + \lambda \cdot \nabla g(\mathbf{x}) \end{cases}$$

Déplacement vers l'attracteur

$$\hat{\mathbf{x}} = \mathbf{x} - k \cdot \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}$$

Recherche de λ

Dans notre cas, la résolution du système donne λ solution de l'équation (intersection de la droite passant par $\hat{\mathbf{x}}$ de vecteur directeur $\nabla g(\mathbf{x})$ et de la fonction de contrainte) :

$$\left(\frac{x^2}{16} + 4y^2\right) \cdot \lambda^2 + \left(\frac{x^2}{4} + 4y^2\right) \cdot \lambda + \left(\frac{x^2}{4} + y^2 - 1\right) = 0$$

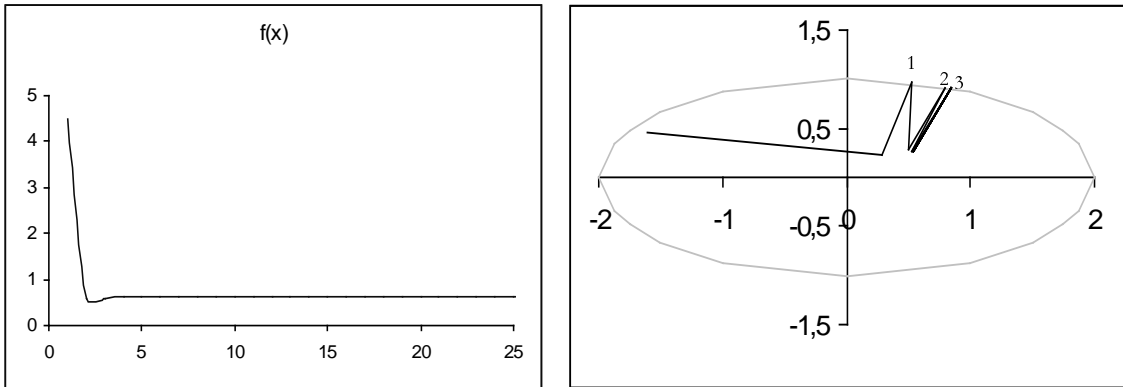
On prend la solution la plus petite car c'est elle qui minimise les déplacements.

Calcul de la nouvelle position

Enfin, on applique la valeur de λ pour trouver la nouvelle position :

$$\mathbf{x}^{i+1} = \hat{\mathbf{x}} + \lambda \cdot \nabla g(\hat{\mathbf{x}}) = \hat{\mathbf{x}} + \lambda \cdot \left\langle \begin{pmatrix} 1 \\ 2 \\ 2 \end{pmatrix}, \hat{\mathbf{x}} \right\rangle$$

Les résultats sont présentés à la Figure 11.

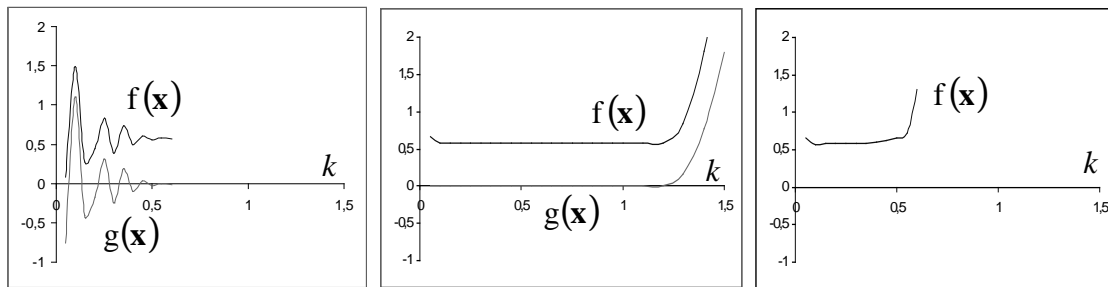


Nombre d'itérations avant stabilisation	11
f(x) final	0,619315116
g(x) final	0

Algorithme de résolution directe

Figure 11. Variations de $f(\mathbf{x})$ et de \mathbf{x} au cours des itérations. La fonction $g(\mathbf{x})$ n'est pas tracée car à la fin de chaque itération, elle est exactement nulle.

2.3.5 Conclusion



(Platt et Barr, 1988)

(Fua et Brechbühler, 1995)

Résolution directe

Figure 12. Évolution des algorithmes en fonction du paramètre k .
Valeurs des fonctions après 25 itérations de l'algorithme.

Il faudrait compléter cette comparaison par une étude théorique pour pouvoir comparer objectivement la qualité de ces trois algorithmes les uns par rapport aux autres. Notons, cependant qu'au vu de cette première comparaison :

- L'algorithme proposé par (Platt et Barr, 1988) oscille autour de la solution avant de se stabiliser. Le système ne converge que très lentement vers la stabilisation. Remarquons cependant que cet algorithme ne fait pas intervenir de position intermédiaire (il nécessite donc moins de calculs par itération, même si le nombre d'itérations nécessaires pour la stabilisation est grand). De plus, la solution finale est toujours très proche de la solution exacte.
- L'algorithme de (Fua et Bechbühler, 1997) est moins sensible au pas de temps (il donne des résultats équivalents quelque soit le pas de temps, voir Figure 12).

Ces deux algorithmes ne permettent pas une résolution exacte de la contrainte (bien que la contrainte soit très bien approchée).

On remarque que l'algorithme présenté dans cette thèse converge assez rapidement mais qu'il fournit, tout en vérifiant exactement la contrainte à chaque itération, une valeur de $f(\mathbf{x})$ moins bonne (voir tableaux de résultats Figure 11 et courbe de la Figure 12). On peut remarquer à la Figure 12 la similitude entre la courbe de l'algorithme de (Fua et Bechbühler, 1997) et de l'algorithme de résolution directe.

La méthode de (Fua et Bechbühler, 1997) présente de nombreux avantages notamment celui d'être plus robuste pas rapport à la valeur du pas de l'algorithme. De plus, elle permet d'éviter des problèmes dus à des sous-espaces de contraintes de même forme que celle présentée à la Figure 6, à droite.

Cette comparaison nous a permis de mieux comprendre le fonctionnement de notre méthode de résolution par rapport à d'autres méthodes développées dans le cadre de manipulation d'objets déformables.

2.4 Multi-contraintes

Dans le cadre de la résolution de plusieurs contraintes, (Platt et Barr, 1988) proposent une généralisation simple. Le problème revient à trouver le point singulier de l'équation suivante :

$$\varepsilon(\mathbf{x}) = f(\mathbf{x}) + \sum_i \alpha_i g_i(\mathbf{x})$$

Dans notre cas, si un élément \mathbf{E}_i est contraint par une contrainte locale et une contrainte globale, la contrainte locale a priorité. Elle est résolue, puis l'élément \mathbf{E}_i est utilisé dans le calcul de la contrainte globale mais n'est pas déplacé. Par exemple, si un élément d'un objet incompressible doit être fixe, l'élément est d'abord déplacé vers la position fixe qu'il doit occuper puis est pris en compte dans la fonction du calcul du volume, mais n'est pas déplacé (les autres éléments se déplaceront donc automatiquement en proportion pour vérifier la contrainte d'incompressibilité).

Dans le cas d'éléments sur-contraints (localement et globalement), des niveaux de priorités doivent être définis. L'algorithme prend d'abord en compte les contraintes de plus forte priorité, puis utilise une méthode itérative pour trouver une solution qui vérifie au mieux les contraintes de faible priorité (cet algorithme itératif est de même type que celui présenté au paragraphe 2.3).

Si plusieurs contraintes ont le même niveau de priorité, on utilise l'intersection des sous-espaces de contraintes comme sous-espace de contraintes. Le vecteur gradient à utiliser pour la projection est le vecteur gradient de cette intersection (ceci n'a pas été expérimenté au cours de cette thèse). Cependant on *ne peut pas* utiliser la méthode de projection dans les trois cas suivants :

- lorsqu'il n'existe pas d'intersection,
- ou lorsque l'intersection ne peut être définie comme une fonction de l'état contraint,
- ou lorsque la fonction définissant l'intersection ne peut être différentiable.

Il faut alors revenir à une méthode itérative (type descente de gradients multiples).

3. Contraintes locales et contraintes globales

Dans ce paragraphe nous allons donner plusieurs exemples de contraintes utilisables dans notre modèle et pouvant être résolus par la méthode de résolution directe. Les contraintes sont soit fixes, soit dynamiques. Dans le premier cas elles doivent toujours être établies pendant la simulation, dans le deuxième cas elles s'établissent sur un signal. On distingue les contraintes locales des contraintes globales.

3.1 Contraintes locales

Les contraintes locales sont des contraintes qui s'appliquent à des éléments isolés. Les contraintes locales les plus utilisées dans nos simulations sont de deux types : fixes ou dynamiques. Par exemple :

- Position fixe : un élément doit rester toujours à la même position dans l'espace euclidien (contrainte fixe).

- Attachement à un pointeur : un pointeur se déplace dans l'espace (rattaché par exemple à un périphérique) et un élément doit toujours se trouver à l'endroit désigné par le pointeur (contrainte fixe).
- Région impénétrable : des régions de l'espace sont définies comme étant impénétrables. Si un quelconque élément viole cet espace, il est déplacé à la périphérie (contrainte dynamique).
- Région forcée : un élément donné doit toujours se trouver à l'intérieur d'une région de l'espace (contrainte fixe).

Résoudre les deux premières contraintes ne pose pas de problème, la projection sur la contrainte ramène l'élément directement à la position où il doit être. Dans le cas d'une contrainte définie par une région, la direction de projection est la direction du gradient de la fonction définissant cette région.

3.2 Contraintes globales

Une contrainte globale met en jeu plusieurs éléments. L'expression de la contrainte se fait à partir des états de tous les éléments mis en jeu. On peut citer des contraintes de déformation, telle que la contrainte de conservation de surface ou de volume. Le paragraphe 4 étudie en détail une contrainte particulière : l'incompressibilité.

Dans notre modèle une contrainte peut être assignée à des éléments de plusieurs régions. Dans ce cas on définit une région pilote (maîtresse) qui contrôle l'apport de chaque région impliquée dans la contrainte (associées). Les régions associées font parvenir les données nécessaires à la région maîtresse qui calcule la contrainte et met en œuvre sa résolution.

4. Une contrainte globale : l'incompressibilité

Au cours du chapitre 1 de ce document, nous avons vu que l'abdomen pouvait être considéré comme une poche d'eau (se déformant à volume constant). L'objectif de ce paragraphe est d'appliquer la méthode de résolution directe à cette contrainte particulièrement utile qu'est l'incompressibilité. Cette contrainte est très importante dans le domaine de l'animation par ordinateur (elle permet de produire des animations d'objets incompressibles) ou même d'améliorer l'aspect de la déformation, mais aussi dans le domaine de la modélisation d'objet et de la conception d'objets assistés par ordinateur (où une déformation à volume constant induit, à densité constante, une préservation de la masse).

La résolution de cette contrainte par notre méthode se révèle être applicable dans de nombreux modèles et est simple à mettre en œuvre.

La méthodologie décrite dans ce paragraphe pour l'application de la contrainte d'incompressibilité est valable pour toute contrainte vérifiant les critères définis au paragraphe 2.2. Les étapes nécessaires sont les suivantes :

- définition de la fonction de contrainte,
- déduction du gradient de la contrainte,
- calcul du système à résoudre, et
- extraction du vecteur de projection.

Nous verrons enfin comment, à partir du résultat obtenu, nous pouvons généraliser notre méthode de déformation à volume constant.

4.1 État de l'art

Il n'existe, à notre connaissance, que trois références donnant une résolution de la contrainte d'incompressibilité : (Platt et Barr, 1988), (Rappoport et al, 1996) et (Aubert et Bechmann, 1995).

Dans (Platt et Barr, 1988), on utilise l'approche des multiplicateurs de Lagrange vue au paragraphe 2.3. La fonction de contrainte est définie comme la différence entre le volume à un instant donné et le volume à conserver.

La méthode de (Rappoport et al, 1996) s'applique à des déformations de solides de Bézier définis par des points de contrôle. C'est le volume local qui est conservé (il n'y a pas de contrôle global du volume). À partir de la position déformée (définie par l'utilisateur), d'une fonction de distance, de contraintes de continuité aux recollements et de la contrainte de volume constant, les auteurs utilisent un algorithme attribué à Uzawa (Ciarlet, 1988) pour trouver une solution. Cet algorithme suit l'approche des multiplicateurs de Lagrange : il minimise la fonction de distance entre la position actuelle et la position déformée tout en maximisant les multiplicateurs de Lagrange des contraintes (c'est-à-dire en se rapprochant de la vérification des contraintes) par une descente de gradients multiples. Le volume et le gradient du volume définis par les objets sont déduits à partir des points de contrôle des solides de Bézier.

(Aubert et Bechmann, 1995), voir aussi (Aubert et Bechmann, 1997), utilisent aussi une méthode de projection, mais itérative, dans un modèle de FFD particulier (le modèle DOGME, voir (Borrel et Bechmann, 1991) et (Bechmann et Dubreuil, 1993)). Les objets sont caractérisés, comme dans notre modèle, par des surfaces triangulées. L'utilisateur définit des déformations au niveau de certains points des objets, et, par interpolation dans l'espace déformé, l'ensemble de l'objet est déformé. Le problème résolu par l'algorithme est la minimisation d'un paramètre décrivant l'écart avec la position désirée, sous la contrainte du volume constant. Pour cela les auteurs utilisent aussi la projection sur l'hyper-plan définie par la contrainte.

On peut aussi citer (Gascuel et Desbrun, 1997) qui apportent une solution particulière à la conservation de volume pendant les déformations utilisant des surfaces implicites (la fonction de potentiel des surfaces implicites est contrôlée par les variations de volume). Cependant cette méthode n'est valable que dans le cas particulier des surfaces implicites.

On le voit toutes les méthodes généralistes décrites dans la littérature se basent sur des méthodes de d'optimisation sous contrainte. (Aubert et Bechmann, 1997) utilise cependant une méthode de projection itérative qui permet de vérifier la contrainte à un seuil d'erreur près.

Une solution triviale pour résoudre la contrainte de volume constant consiste à utiliser une homothétie. En appliquant cette opération par rapport à l'isobarycentre d'un objet et en utilisant le rapport des volumes comme paramètre de la transformation, on obtient bien une déformation iso-volumique. Cependant cette solution n'est valable que pour des objets géométriques simples et réguliers. Dans des cas plus complexes, l'homothétie engendre des

déformations supplémentaires brutales ou visuellement incorrectes (des retournements de surfaces peuvent même apparaître).

Nous allons maintenant montrer comment utiliser la méthode de résolution directe pour une contrainte globale comme la contrainte de volume constant.

4.2 Fonction de volume

Pour appliquer la méthode de résolution directe nous devons expliciter la fonction de contrainte $C(\mathbf{x})=0$. La contrainte d'incompressibilité s'exprime de la manière suivante :

$$C(\mathbf{x})=V(\mathbf{x})-V^0,$$

où V^0 est la valeur du volume à conserver, $V(\mathbf{x})$ donne le volume de l'objet, avec \mathbf{x} le vecteur de dimension $3n$ correspondant à la position des éléments composant cet objet (tous les éléments de l'objet définissent sa surface). C'est une contrainte globale sur l'ensemble de ces n éléments.

Pour calculer $V(\mathbf{x})$, on part du théorème d'Ostrogradsky (on pourra aussi se référer à (Goldman, 1991) et (Mantyla, 1988) pour une expression discrète du volume). Ce théorème énonce que l'intégrale de la divergence d'un champ de vecteurs, étendue à un volume est égale au flux de ce vecteur sortant de la surface qui limite ce volume :

$$\iiint_{\tau} \text{div}(\mathbf{u}) d\tau = \iint_S \langle \mathbf{n}, \mathbf{u} \rangle d\sigma,$$

où S est la surface fermée limitant le volume τ , \mathbf{n} le vecteur unitaire de la normale (un sens positif ayant été fixé), $d\tau$ un élément de volume et $d\sigma$ un élément de surface. On pose la paramétrisation suivante :

$$\mathbf{u} = (u_x(x, y, z) \quad u_y(x, y, z) \quad u_z(x, y, z)),$$

$$\text{d'où } \text{div}(\mathbf{u}) = \frac{\partial u_x}{\partial x} + \frac{\partial u_y}{\partial y} + \frac{\partial u_z}{\partial z}.$$

Si on choisit \mathbf{u} tel que $\text{div}(\mathbf{u})=1$, c'est-à-dire que :

$$\frac{\partial u_x}{\partial x} + \frac{\partial u_y}{\partial y} + \frac{\partial u_z}{\partial z} = 1.$$

Alors on obtient :

$$\iiint_{\tau} \text{div}(\mathbf{u}) = \iiint_{\tau} d\tau = V.$$

Pour cela on peut prendre $u_x(x, y, z) = \frac{1}{3}x$, $u_y(x, y, z) = \frac{1}{3}y$ et $u_z(x, y, z) = \frac{1}{3}z$, soit :

$$\mathbf{u} = \frac{1}{3}(x \quad y \quad z)$$

Soit $\mathbf{p} = (x \quad y \quad z)$ un point de S , on a finalement :

$$V = \iint_S \langle \mathbf{n}, \mathbf{u} \rangle d\sigma = \frac{1}{3} \iint_S \langle \mathbf{n}, \mathbf{p} \rangle d\sigma$$

Dans notre modèle la surface S d'un objet est constituée de k facettes.

Pour une facette F_k , on pose :

$$\text{Surface}(F_k) = \iint_{F_k} d\sigma,$$

où $d\sigma$ parcourt cette fois-ci la surface de la facette F_k .

on a donc :

$$\begin{aligned} V &= \frac{1}{3} \sum_k \iint_{F_k} \langle \mathbf{p}, \mathbf{n}_k \rangle d\sigma \\ &= \frac{1}{3} \sum_k \iint_{F_k} \langle \mathbf{p} d\sigma, \mathbf{n}_k \rangle, \\ &= \frac{1}{3} \sum_k \langle \mathbf{n}_k, \iint_{F_k} \mathbf{p} d\sigma \rangle \end{aligned}$$

où F_k est la $k^{\text{ième}}$ facette et \mathbf{n}_k son vecteur normal unitaire.

Comme on l'a dit au chapitre précédent, on a choisi des facettes triangulaires comme représentation de la surface parce que cela permettait, entre autres, de simplifier les calculs du volume. En effet, soit \mathbf{g}_k l'isobarycentre de la facette k , on a :

$$\mathbf{g}_k = \frac{\iint_{F_k} \mathbf{p} d\sigma}{\iint_{F_k} d\sigma} = \frac{\iint_{F_k} \mathbf{p} d\sigma}{\text{Surface}(F_k)}, \text{ c'est-à-dire que } \iint_{F_k} \mathbf{p} d\sigma = \text{Surface}(F_k) \cdot \mathbf{g}_k.$$

On obtient donc, après simplification :

$$V = \frac{1}{3} \sum_k \text{Surface}(F_k) \cdot \langle \mathbf{g}_k, \mathbf{n}_k \rangle.$$

C'est l'expression du volume en fonction de chacune des facettes. Dans (Promayon et al, 1996) tous les calculs étaient basés sur cette expression. Cela donne une certaine complexité à l'expression du gradient et donc au calcul de la contrainte.

En fait, il est beaucoup plus simple d'exprimer le volume (et le gradient du volume) non plus par rapport aux facettes, mais par rapport aux éléments eux-mêmes. C'est ce que nous allons voir maintenant.

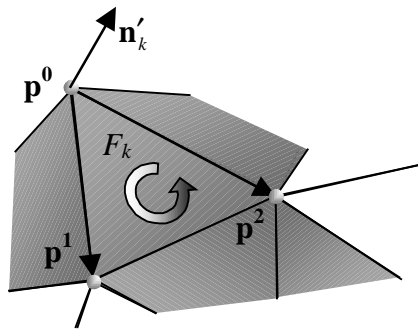


Figure 13. La facette F_k est composée de trois points définis dans le sens trigonométrique.

Comme F_k est un triangle composé par les trois éléments de coordonnées $(\mathbf{p}_k^0 \ \mathbf{p}_k^1 \ \mathbf{p}_k^2)$, (voir figure ci-dessus) on a :

$$\text{Surface}(F_k) = \frac{1}{2} \|\mathbf{n}'_k\|,$$

avec \mathbf{n}'_k la normale à la facette F_k défini par :

$$\mathbf{n}'_k = \mathbf{p}_k^0 \mathbf{p}_k^1 \times \mathbf{p}_k^0 \mathbf{p}_k^2, \text{ ainsi : } \mathbf{n}_k = \frac{\mathbf{n}'_k}{\|\mathbf{n}'_k\|}.$$

Finalement :

$$\begin{aligned} V &= \frac{1}{3} \sum_k \frac{1}{2} \|\mathbf{n}'_k\| \cdot \langle \mathbf{g}_k, \frac{\mathbf{n}'_k}{\|\mathbf{n}'_k\|} \rangle \\ &= \frac{1}{6} \sum_k \langle \mathbf{g}_k, \mathbf{n}'_k \rangle \end{aligned}$$

Ceci n'est toujours pas une expression du volume en fonction des coordonnées des éléments $(\mathbf{p}_k^0 \ \mathbf{p}_k^1 \ \mathbf{p}_k^2)$.

On remarque que $\mathbf{n}'_k = \mathbf{p}_k^0 \mathbf{p}_k^1 \times \mathbf{p}_k^0 \mathbf{p}_k^2$ et $\mathbf{g}_k = \frac{1}{3}(\mathbf{p}_k^0 + \mathbf{p}_k^1 + \mathbf{p}_k^2)$, d'où :

$$\begin{aligned} V &= \frac{1}{6} \sum_k \langle \frac{1}{3}(\mathbf{p}_k^0 + \mathbf{p}_k^1 + \mathbf{p}_k^2), \mathbf{n}'_k \rangle \\ &= \frac{1}{18} \left(\sum_k \langle \mathbf{p}_k^0, \mathbf{n}'_k \rangle + \sum_k \langle \mathbf{p}_k^1, \mathbf{n}'_k \rangle + \sum_k \langle \mathbf{p}_k^2, \mathbf{n}'_k \rangle \right) \end{aligned}$$

Maintenant, on peut extirper de la formule ci-dessus, l'ensemble des coordonnées $(\mathbf{p}_1 \cdots \mathbf{p}_n)$ des n éléments composant l'objet. On regroupe les produits scalaires en factorisant par les \mathbf{p}_i . Pour plus de clarté on renomme maintenant les éléments en ne considérant plus leur ordre dans les facettes mais leur indice global dans l'objet.

On obtient :

$$(1) \quad V = \frac{1}{18} \left(\sum_{j=1}^{nbf(1)} \langle \mathbf{p}_1, \mathbf{n}'_{F(1,j)} \rangle + \sum_{j=1}^{nbf(2)} \langle \mathbf{p}_2, \mathbf{n}'_{F(2,j)} \rangle + \cdots + \sum_{j=1}^{nbf(n)} \langle \mathbf{p}_n, \mathbf{n}'_{F(n,j)} \rangle \right),$$

où $nbf(i)$ est le nombre de facettes concourantes en \mathbf{p}_i , $F(i, j)$ la $j^{\text{ème}}$ facette concourante en \mathbf{p}_i (donc $\mathbf{n}'_{F(i,j)}$ est la normale de la facette $F(i, j)$). La formule (1) est la sommation de la contribution au volume de chacun des n éléments constituant l'objet :

$$V = \frac{1}{18} \sum_{i=1}^n V_i = \frac{1}{18} \sum_{i=1}^n \sum_{j=1}^{nbf(i)} \langle \mathbf{p}_i, \mathbf{n}'_{F(i,j)} \rangle.$$

Examinons la contribution de l'élément \mathbf{p}_i :

$$V_i = \sum_{j=1}^{nbf(i)} \langle \mathbf{p}_i, \mathbf{n}'_{F(i,j)} \rangle = \langle \mathbf{p}_i, \sum_j \mathbf{n}'_{F(i,j)} \rangle.$$

Or on a (voir Figure 14) :

$$\begin{aligned} \mathbf{n}'_{F(i,j)} &= \mathbf{p}_i \mathbf{p}_{v(i,j)} \times \mathbf{p}_i \mathbf{p}_{v(i,j \oplus 1)} = (\mathbf{p}_{v(i,j)} - \mathbf{p}_i) \times (\mathbf{p}_{v(i,j \oplus 1)} - \mathbf{p}_i) \\ &= \mathbf{p}_{v(i,j)} \times \mathbf{p}_{v(i,j \oplus 1)} + \mathbf{p}_i \times \mathbf{p}_{v(i,j)} \mathbf{p}_{v(i,j \oplus 1)} \end{aligned}$$

avec $v(i,j)$ désignant le $j^{\text{ème}}$ voisin de l'élément et \oplus désignant l'addition modulo le $nbf(i)$.

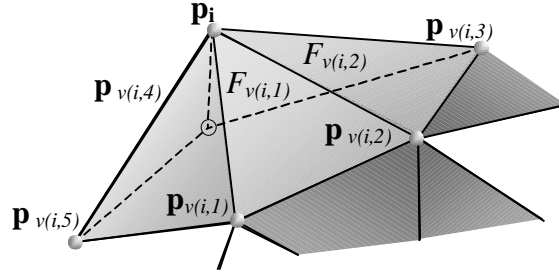


Figure 14. Disposition des voisins d'un élément en \mathbf{p}_i . L'élément i a autant de voisins que de facettes concourantes car ces dernières sont triangulaires.

Quand on fait la sommation sur l'ensemble des facettes concourantes :

$$\sum_{j=1}^{nbf(i)} \mathbf{p}_i \times \mathbf{p}_{v(i,j)} \mathbf{p}_{v(i,j \oplus 1)} = \mathbf{p}_i \times \sum_{j=1}^{nbf(i)} \mathbf{p}_{v(i,j)} \mathbf{p}_{v(i,j \oplus 1)} = \mathbf{p}_i \times \mathbf{0} = \mathbf{0},$$

où $\mathbf{0}$ dénote le vecteur nul. On en déduit que :

$$(2) \quad \mathbf{n}''_i = \sum_j \mathbf{n}'_{F(i,j)} = \sum_j \mathbf{p}_{v(i,j)} \times \mathbf{p}_{v(i,j \oplus 1)}.$$

\mathbf{n}''_i est une expression de la normale en \mathbf{p}_i (voir annexe 3).

On obtient finalement l'expression du volume en fonction des éléments :

$$V = \frac{1}{18} \sum_i V_i = \frac{1}{18} \sum_i \langle \mathbf{p}_i, \mathbf{n}''_i \rangle.$$

Nous allons maintenant chercher le gradient de ce volume.

4.3 Le gradient de la contrainte

On regarde ce qui se passe pour un élément :

$$\nabla V_i = \frac{\partial V_i}{\partial \mathbf{p}_i} = \frac{\partial \langle \mathbf{p}_i, \mathbf{n}''_i \rangle}{\partial \mathbf{p}_i} = \frac{\partial \mathbf{p}_i}{\partial \mathbf{p}_i} \cdot \mathbf{n}''_i + \mathbf{p}_i \cdot \frac{\partial \mathbf{n}''_i}{\partial \mathbf{p}_i}.$$

On a vu à l'équation (2) que \mathbf{n}''_i est indépendant de \mathbf{p}_i . On en déduit : $\nabla V_i = \mathbf{n}''_i$.

Ainsi (voir Figure 15 et annexe 3) :

LE GRADIENT DE LA FONCTION DE VOLUME POUR UN ÉLÉMENT EST COLINÉAIRE À LA NORMALE \mathbf{n}''_i .

Ce résultat est particulièrement important puisqu'il permet de généraliser la contrainte d'incompressibilité à d'autres modèles (voir paragraphe 4.5). Le lecteur pourra se référer à (Promayon et al, 1996) pour l'expression du volume et du gradient en terme des facettes.

4.4 Le vecteur de projection

D'après la méthode de résolution directe le système à résoudre est finalement le suivant :

$$\begin{cases} \mathbf{x}^{t+dt} = \widehat{\mathbf{x}}^{t+dt} + \lambda \cdot \nabla C(\widehat{\mathbf{x}}^{t+dt}) \\ C(\mathbf{x}) = 0 \end{cases} \Leftrightarrow \begin{cases} \mathbf{x}^{t+dt} = \widehat{\mathbf{x}}^{t+dt} + \lambda \cdot \nabla C(\widehat{\mathbf{x}}^{t+dt}) \\ V(\mathbf{x}^{t+dt}) - V^0 = 0 \end{cases},$$

où $\widehat{\mathbf{x}}^{t+dt}$ est la position atteinte sous l'action des forces sans tenir compte des contraintes (position intermédiaire).

Comme le vecteur \mathbf{x}^{t+dt} est de dimension $3n$ si l'objet incompressible est constitué de n éléments, ce système a $3n+1$ équations et $3n+1$ inconnues (\mathbf{x}^{t+dt} et λ). Pour simplifier le raisonnement et les calculs, on se place au niveau de chaque élément. On cherche $\mathbf{x}^{t+dt} = (\mathbf{p}_1 \cdots \mathbf{p}_n)$ et λ . Pour chaque élément, on a :

$$(3) \quad \begin{cases} \mathbf{p}_i^{t+dt} = \widehat{\mathbf{p}}_i^{t+dt} + \lambda \cdot \widehat{\mathbf{n}}_i^{t+dt} \\ V(\mathbf{x}^{t+dt}) - V^0 = 0 \end{cases} \Leftrightarrow \begin{cases} \mathbf{p}_i^{t+dt} = \widehat{\mathbf{p}}_i^{t+dt} + \lambda \cdot \widehat{\mathbf{n}}_i^{t+dt} \\ \frac{1}{18} \sum_{i=1}^n \langle \mathbf{p}_i^{t+dt}, \mathbf{n}_i^{t+dt} \rangle - V^0 = 0 \end{cases},$$

où $\widehat{\mathbf{p}}_i^{t+dt}$ est la position du $i^{\text{ème}}$ élément de l'objet après application des forces et sans tenir compte des contraintes (c'est la position intermédiaire). $\widehat{\mathbf{n}}_i^{t+dt}$ est normale à la surface en $\widehat{\mathbf{p}}_i^{t+dt}$.

Remarquons ici que c'est le volume global qui reste constant et non les volumes locaux contrôlés par les éléments. Pour conserver les volumes locaux, il faudrait utiliser les n systèmes définis par (i parcourant les n éléments) :

$$\begin{cases} \mathbf{p}_i^{t+dt} = \widehat{\mathbf{p}}_i^{t+dt} + \lambda_i \cdot \widehat{\mathbf{n}}_i^{t+dt} \\ \frac{1}{18} \langle \mathbf{p}_i^{t+dt}, \mathbf{n}_i^{t+dt} \rangle - V_i^0 = 0 \end{cases},$$

où V_i^0 est le volume local à conserver.

Nous allons maintenant calculer l'équation en λ de l'intersection entre le sous-espace de la contrainte et la droite de vecteur directeur le gradient de la contrainte.

À partir du système (3), on a :

$$\begin{cases} \mathbf{p}_i^{t+dt} = \widehat{\mathbf{p}}_i^{t+dt} + \lambda \cdot \widehat{\mathbf{n}}_i^{t+dt} \\ \frac{1}{18} \sum_{i=1}^n \langle \mathbf{p}_i^{t+dt}, \mathbf{n}_i^{t+dt} \rangle - V^0 = 0 \end{cases} \\ \Leftrightarrow \frac{1}{18} \sum_{i=1}^n \langle \widehat{\mathbf{p}}_i^{t+dt} + \lambda \cdot \widehat{\mathbf{n}}_i^{t+dt}, \mathbf{n}_i^{t+dt} \rangle - V^0 = 0$$

Développons la formule en chaque élément :

$$\begin{aligned}
 \langle \widehat{\mathbf{p}}_i^{t+dt} + \lambda \cdot \widehat{\mathbf{n}}_i^{\prime\prime t+dt}, \mathbf{n}_i^{\prime\prime t+dt} \rangle &= \langle \widehat{\mathbf{p}}_i^{t+dt} + \lambda \cdot \widehat{\mathbf{n}}_i^{\prime\prime t+dt}, \sum_{j=1}^{nbf(i)} \mathbf{p}_{v(i,j)}^{t+dt} \times \mathbf{p}_{v(i,j\oplus 1)}^{t+dt} \rangle \\
 &= \langle \widehat{\mathbf{p}}_i^{t+dt} + \lambda \cdot \widehat{\mathbf{n}}_i^{\prime\prime t+dt}, \sum_{j=1}^{nbf(i)} (\widehat{\mathbf{p}}_{v(i,j)}^{t+dt} + \lambda \cdot \widehat{\mathbf{n}}_{v(i,j)}^{\prime\prime t+dt}) \times (\widehat{\mathbf{p}}_{v(i,j\oplus 1)}^{t+dt} + \lambda \cdot \widehat{\mathbf{n}}_{v(i,j\oplus 1)}^{\prime\prime t+dt}) \rangle \\
 &= \langle \widehat{\mathbf{p}}_i^{t+dt} + \lambda \cdot \widehat{\mathbf{n}}_i^{\prime\prime t+dt}, \sum_{j=1}^{nbf(i)} (\widehat{\mathbf{p}}_{v(i,j)}^{t+dt} \times \widehat{\mathbf{p}}_{v(i,j\oplus 1)}^{t+dt} \\
 &\quad + \lambda \cdot (\widehat{\mathbf{n}}_{v(i,j)}^{\prime\prime t+dt} \times \widehat{\mathbf{p}}_{v(i,j\oplus 1)}^{t+dt} + \widehat{\mathbf{p}}_{v(i,j)}^{t+dt} \times \widehat{\mathbf{n}}_{v(i,j\oplus 1)}^{\prime\prime t+dt}) \\
 &\quad + \lambda^2 \cdot \widehat{\mathbf{n}}_{v(i,j)}^{\prime\prime t+dt} \times \widehat{\mathbf{n}}_{v(i,j\oplus 1)}^{\prime\prime t+dt}) \rangle \\
 &= \langle \widehat{\mathbf{p}}_i^{t+dt}, \sum_j \widehat{\mathbf{p}}_{v(i,j)}^{t+dt} \times \widehat{\mathbf{p}}_{v(i,j\oplus 1)}^{t+dt} \rangle \\
 &\quad + \lambda \cdot \left(\langle \widehat{\mathbf{n}}_i^{\prime\prime t+dt}, \sum_j \widehat{\mathbf{p}}_{v(i,j)}^{t+dt} \times \widehat{\mathbf{p}}_{v(i,j\oplus 1)}^{t+dt} \rangle \right. \\
 &\quad \left. + \langle \widehat{\mathbf{p}}_i^{t+dt}, \sum_j (\widehat{\mathbf{n}}_{v(i,j)}^{\prime\prime t+dt} \times \widehat{\mathbf{p}}_{v(i,j\oplus 1)}^{t+dt} + \widehat{\mathbf{p}}_{v(i,j)}^{t+dt} \times \widehat{\mathbf{n}}_{v(i,j\oplus 1)}^{\prime\prime t+dt}) \rangle \right) \\
 &\quad + \lambda^2 \cdot \left(\langle \widehat{\mathbf{n}}_i^{\prime\prime t+dt}, \sum_j (\widehat{\mathbf{n}}_{v(i,j)}^{\prime\prime t+dt} \times \widehat{\mathbf{p}}_{v(i,j\oplus 1)}^{t+dt} + \widehat{\mathbf{p}}_{v(i,j)}^{t+dt} \times \widehat{\mathbf{n}}_{v(i,j\oplus 1)}^{\prime\prime t+dt}) \rangle \right) \\
 &\quad \left. + \langle \widehat{\mathbf{p}}_i^{t+dt}, \sum_j \widehat{\mathbf{n}}_{v(i,j)}^{\prime\prime t+dt} \times \widehat{\mathbf{n}}_{v(i,j\oplus 1)}^{\prime\prime t+dt} \rangle \right) \\
 &\quad + \lambda^3 \cdot \langle \widehat{\mathbf{n}}_i^{\prime\prime t+dt}, \sum_j \widehat{\mathbf{n}}_{v(i,j)}^{\prime\prime t+dt} \times \widehat{\mathbf{n}}_{v(i,j\oplus 1)}^{\prime\prime t+dt} \rangle \\
 &= d_i + c_i \cdot \lambda + b_i \cdot \lambda^2 + a_i \cdot \lambda^3
 \end{aligned}$$

On a donc trouvé la contribution a_i b_i c_i et d_i de chaque élément à l'équation globale d'intersection.

On peut remarquer que $d_i = \widehat{V}_i^{t+dt}$, donc $\sum_i d_i = \widehat{V}^{t+dt}$ volume de l'objet déformé.

Le traitement de la contrainte de volume constant peut être fait par un algorithme parallèle puisque les calculs se font au niveau de chaque élément. Il suffit de résoudre maintenant :

$$\begin{aligned}
 &\frac{1}{18} \sum_{i=1}^n \langle \widehat{\mathbf{p}}_i^{t+dt} + \lambda \cdot \widehat{\mathbf{n}}_i^{\prime\prime t+dt}, \mathbf{n}_i^{\prime\prime t+dt} \rangle - V^0 = 0 \\
 &\Leftrightarrow \sum_{i=1}^n (a_i \cdot \lambda^3 + b_i \cdot \lambda^2 + c_i \cdot \lambda + d_i) - 18 \cdot V^0 = 0 \\
 (4) \quad &\Leftrightarrow \lambda^3 \sum_{i=1}^n a_i + \lambda^2 \sum_{i=1}^n b_i + \lambda \sum_{i=1}^n c_i + \sum_{i=1}^n d_i - 18 \cdot V^0 = 0
 \end{aligned}$$

C'est une équation du 3^{ième} degré classique, dont on va prendre la racine réelle de valeur absolue minimale. Le choix de la plus petite valeur absolue λ_{\min} garantit le déplacement minimum le long du gradient.

En injectant λ_{\min} , plus petite solution de (4), dans le système (3), \mathbf{x}^{t+dt} (position vérifiant la contrainte d'incompressibilité) est obtenu en une seule passe.

Pour les éléments contraints (fixés dans l'espace, contraints à la position d'un autre élément ou à la position interactivement manipulé par l'utilisateur), le λ correspondant est mis à zéro. Ainsi l'élément participe bien au calcul du volume, mais n'est pas déplacé.

Le déplacement final d'un élément de sa position intermédiaire, avant la vérification des contraintes, $\hat{\mathbf{p}}_i^{t+dt}$ vers la solution est $\lambda \cdot \hat{\mathbf{n}}_i^{t+dt}$.

4.5 Généralisation

Pour conclure ce paragraphe, nous allons tenter de généraliser la résolution de la contrainte d'incompressibilité en nous appuyant sur le fait que le gradient de la fonction de volume pour un élément est colinéaire à la normale à la surface en cet élément.

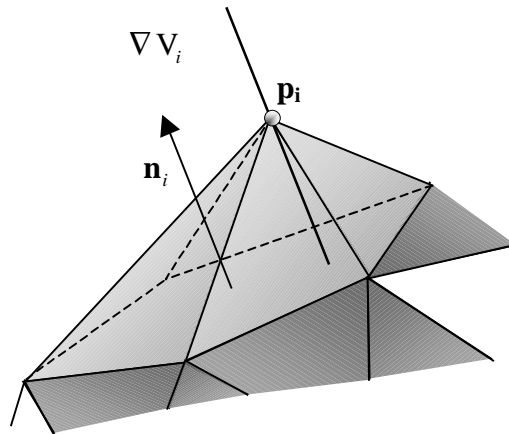


Figure 15. Le gradient de la fonction de volume est colinéaire à la normale en un point de la triangulation.

4.5.1 Validité physique

En projetant les éléments sur le sous-espace de la contrainte, on les déplace selon le gradient de la fonction de contrainte. Dans notre optique le déplacement dû à la contrainte provient des forces engendrées, dans la réalité, par les contraintes. En général ces forces ont un module infini (ou quasi-infini). Lorsque deux éléments sont fixés l'un à l'autre, par exemple, aucune force extérieure ne sera capable pendant la simulation de séparer ces éléments. Au contraire des méthodes de contrainte de réaction, dans notre méthode, on choisit de ne pas exprimer explicitement ces forces mais de calculer directement les déplacements dus à celle-ci.

En ce qui concerne la contrainte d'incompressibilité, le gradient est colinéaire à une normale définie à la surface au niveau des éléments (bien que cette normale n'existe pas mathématiquement, on choisit de prendre la moyenne des normales des facettes concourantes au niveau des éléments). Le déplacement des éléments se fera donc selon la

normale à la surface. Or on sait que les forces de pression (liées au volume des objets) sont colinéaires à la normale à la surface. On peut déduire que nos éléments sont soumis à des forces dues à la pression et permettant de préserver le volume. L'action de ces forces se traduit directement dans la résolution de la contrainte par des déplacements des éléments selon la normale à la surface.

4.5.2 Compressibilité

En résolvant la contrainte, on a trouvé, à partir d'un état déformé, un état déformé mais conservant le volume de départ. On peut donc modéliser une matière partiellement incompressible en utilisant l'état incompressible comme attracteur (c'est-à-dire en créant une force entre l'état déformé et l'état déformé de volume contraint). À la connaissance de l'auteur, cette propriété physique n'a jamais été directement modélisée, puisque aucune méthode précédente ne détermine avec exactitude l'état incompressible.

D'autre part, on peut faire varier le volume d'un objet au cours d'une simulation en changeant la valeur V^0 dans la résolution de la contrainte.

De plus, on l'a vu, on peut conserver le volume non plus globalement, mais localement.

4.5.3 Autres modèles

La résolution directe peut, bien évidemment, être appliquée à d'autres modèles que le nôtre (voire d'autres méthodes de résolution, comme dans des algorithmes de minimisation classique, paragraphe 2.3). Il en va de même avec la contrainte d'incompressibilité.

Un algorithme simple peut être utilisé avec d'autres types de modèles (surface facettisée ou non, triangles ou non, basé sur la physique ou non). Une fois l'objet déformé, les éléments de surface doivent être déplacés chacun selon la direction de leur normale jusqu'à ce que le volume soit identique au volume initial. Cet algorithme est illustré à la Figure 16.

Pour utiliser notre méthode il faut que :

- Soit les éléments de surface peuvent être manipulés et déplacés directement, dans ce cas la méthode présentée ici est entièrement valable,
- Soit la surface est manipulée par l'intermédiaire de points de contrôle. Dans ce cas il faut connaître la relation entre un déplacement d'un point de contrôle et la variation de volume entraînée par ce déplacement. C'est le cas par exemple des carreaux ou volumes de Bézier. Dans ce cas l'équation du 3^{ème} degré (4) permettant de trouver la valeur de λ sera différente, mais méthode de résolution reste la même.

Il est possible d'utiliser notre méthode avec les volumes de Bézier de (Rappoport et al, 1996). L'algorithme de minimisation utilisé ne doit plus prendre en compte la résolution de la contrainte d'incompressibilité puisque celle-ci peut-être déduite à partir de n'importe quel état. Il serait intéressant de comparer les résultats avec ceux déjà obtenus.

La méthode de résolution directe par projection (en une seule passe) a été intégrée par F. Aubert dans le modèle DOGME (qui est de type FFD), au Laboratoire des Sciences de l'Image, de l'Informatique et de la Télédétection de Strasbourg. Le terme d'optimisation du modèle DOGME permet cette intégration et l'obtention de bons résultats visuels.

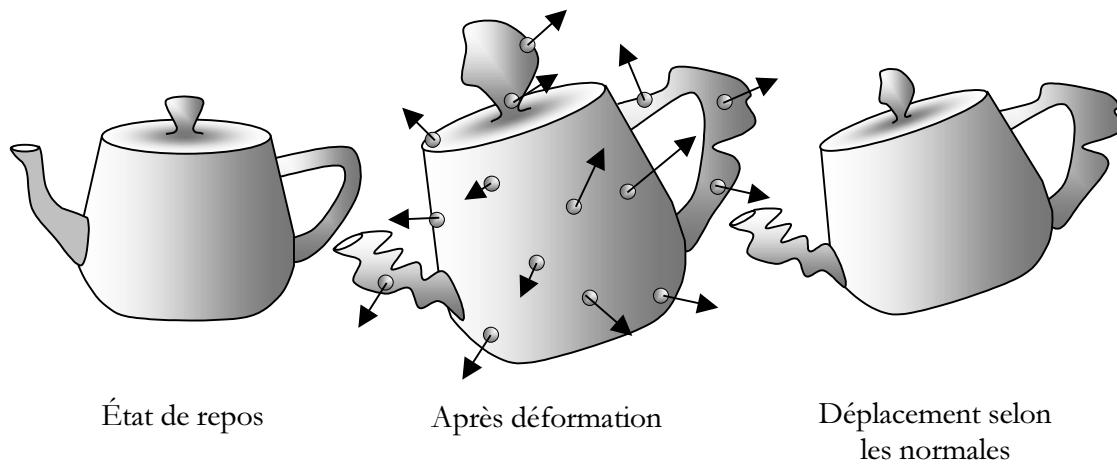


Figure 16. Algorithme simple : une fois l'objet déformé, les éléments de surfaces doivent être déplacés chacun selon la direction de leur normale jusqu'à ce que le volume soit identique au volume initial.

5. Conclusion

Dans ce chapitre nous avons vu comment les raccordements entre les surfaces pouvaient être modélisés. Puis nous avons examiné de près la méthode de résolution directe pour vérifier les contraintes. Enfin, nous avons donné un exemple de l'application de cette méthode avec la contrainte d'incompressibilité.

Notre modèle informatique est maintenant complet et nous verrons des exemples de son utilisation dans la dernière partie de ce manuscrit. Ces exemples serviront à la validation qualitative du modèle.

Chapitre 5

Simulation

[II] sait que ce n'est pas avec des recettes qu'on touche à l'essence d'un art [...] La technique est un domaine creux et méprisable quand elle est fermée sur elle-même.

Bernard Tirtiaux, Le passeur de lumière.

L'étude de la physiologie nous a montré que la simulation des mouvements respiratoires du tronc pouvait être considérée comme la simulation d'un objet hétérogène composé de régions ayant des propriétés différentes. Dans la partie précédente nous avons construit un modèle informatique capable de gérer des objets hétérogènes dans le cadre d'une simulation. Ce modèle manipule un système dynamique ayant des propriétés basées sur la physique. Ces propriétés ont été modélisées soit directement, soit par l'intermédiaire de contraintes locales ou globales.

Nous allons maintenant utiliser ce modèle dans des exemples afin de mieux comprendre son fonctionnement. L'utilisation d'exemples passe par la définition des objets hétérogènes que le modèle manipule. Parmi ces objets, on construira, en fin de chapitre, l'objet permettant de modéliser le comportement du tronc. On parlera alors de « modèle du tronc » dans le sens où cet objet représente le tronc et ses propriétés.

Enfin, nous construirons le modèle du tronc.

Le paragraphe Chapitre 11 décrit plus en détail l'implémentation informatique. Le paragraphe Chapitre 12 présente quelques exemples relativement simples qui permettent de vérifier certaines des hypothèses contenues dans le modèle informatique. Le paragraphe Chapitre 13 décrit plus particulièrement la construction du modèle du tronc, dans lequel la contraction diaphragmatique engendre les mouvements, et sa validation qualitative. Enfin le dernier paragraphe met en place les éléments de comparaison et de validation quantitative d'un modèle « réaliste » du tronc.

1. Implémentation

Parmi les objectifs initiaux concernant l'utilisation du modèle lors d'une simulation, nous avons insisté sur des caractéristiques techniques telles que la portabilité, la rapidité et la facilité de manipulation. Nous verrons au cours de ce paragraphe que la démarche d'implémentation du modèle en C++ suit ces objectifs.

1.1 Développement et interface

Après une rapide vérification bi-dimensionnelle, une première version du modèle a été implémentée sous PEX (Gaskins, 1992) pour le graphique 3D et libsx¹ pour l'interface 2D. OpenGL, (Neider et al., 1993) et (Kilgard, 1996), ayant entre temps supplanté PEX dans le monde des bibliothèques graphiques 3D, la version suivante du modèle a été développée entièrement en C++ en utilisant OpenGL pour la 3D et V² pour la 2D. V et OpenGL (par l'intermédiaire de Mesa³) sont disponibles sous X11 sur tous les systèmes Unix (Linux compris) mais aussi sous Microsoft Windows. Cela permet d'assurer une grande portabilité. L'implémentation utilise une bibliothèque d'algèbre linéaire, développée au laboratoire par E. Bainville, toujours en C++ et nommée « ALP ».

L'implémentation de la version actuelle représente environ 7000 lignes de code pour l'interface homme-machine, et 5000 lignes de code pour la bibliothèque du modèle proprement dit. Cette bibliothèque comprend les classes (données et méthodes d'accès) correspondant au modèle décrit dans les deux chapitres précédents comme nous allons le voir au paragraphe suivant.

1.2 Implémentation du modèle

Nous détaillons un peu ici l'implémentation de la bibliothèque du modèle.

1.2.1 Architecture

L'architecture de la bibliothèque suit l'approche générale du modèle décrit au chapitre 3 (Figure 1). L'architecture de notre bibliothèque est donnée à la Figure 1 (partie indépendante de l'affichage et de la manipulation).

1.2.2 Données

Nous avons choisi de représenter les surfaces sous formes de triangles. Cela nous permet d'importer des données de différents formats provenant de modeleurs classiques de CAO. Plusieurs filtres ont été implémentés permettant d'importer des fichiers de données des principaux formats d'objets 3D (Renderman, 3D Studio, openInventor, VRML) lorsque ceux-ci sont composés de facettes triangulaires. Un filtre générique permettant de construire un fichier de données à partir d'une liste de points et de triangles sert d'intermédiaire entre ces

¹ <http://zeus.chem.wvu.edu/libsx/libsx.html>

² <http://www.cs.unm.edu/~wampler/vgui/vgui.html>

³ <http://www.ssec.wisc.edu/~brianp/Mesa.html>

formats et le nôtre. Les fichiers provenant d'algorithmes de segmentation par δ -snakes (Bainville, 1992) sont utilisables.

D'autres fonctions ont été implémentées permettant de changer la position, l'orientation ou l'échelle des objets, ainsi que de concaténer plusieurs objets. L'annexe 4 fournit une description complète du format de données que nous avons créé.

Les objets utilisables par notre modèle peuvent ainsi être créés à partir d'un grand nombre de bases de données d'objets 3D.

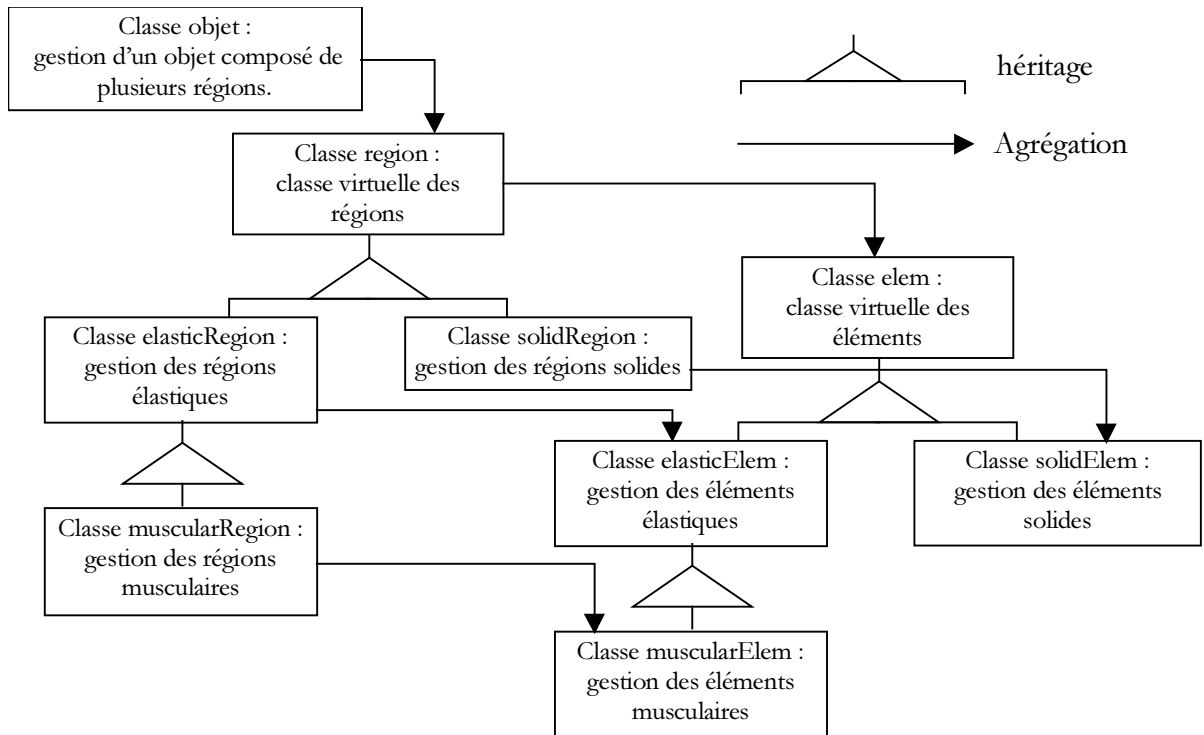


Figure 1. Architecture de la librairie orienté objet (C++) du modèle.

1.2.3 Manipulation

L'interface développée en C++ avec la librairie V (nous y avons développé une classe spéciale de manipulation sous OpenGL) nous permet de manipuler et d'accéder aux informations dynamiques des éléments ou des régions au cours de la simulation. Dans la première version de la librairie la manipulation des éléments pouvait se faire directement par l'intermédiaire d'une souris à 6 degrés de libertés (Flock of Bird® d'Ascension Technology™). Un ou plusieurs éléments pouvaient être sélectionnés et manipulés, la souris 6D donnant la position et l'orientation à chaque pas de temps. La camera pouvait aussi être manipulée par l'intermédiaire de la souris 6D. Cette étude a fait l'objet d'un stage d'IUT (Simon, 1996) résolvant un certain nombre de problèmes communs à toute application de réalité virtuelle

(manipulation directe d'objet en environnement virtuel, obtention de mouvements naturels et acquisition de données en temps réel, appréciation et représentation de la 3^{ème} dimension et précision du périphérique).

Comme on le verra au paragraphe 2.3, la manipulation peut se faire tout en laissant tourner la simulation (les forces élastiques permettent alors de lisser la surface).

L'implémentation du modèle a été guidée par les mêmes objectifs que sa conception. Nous allons voir au paragraphe suivant que l'étape de test et de validation en est facilitée.

2. Exemples

Ce paragraphe montre quelques exemples d'utilisation du modèle permettant de mieux comprendre le fonctionnement du modèle et d'en vérifier certaines caractéristiques. Pour chaque exemple nous décrirons les objectifs, l'objet utilisé et les temps de calculs correspondants. Les deux premiers exemples utilisent simplement le modèle tel qu'il a été présenté aux chapitres précédents. Le troisième exemple montre comment la résolution de la contrainte du volume constant permet la modélisation d'objet à volume constant.

2.1 Deux ballons connectés

Cet exemple est tiré de (Promayon et al, 1996) et est inspiré de (Aubert et Bechmann, 1995). Il montre comment s'articule l'ensemble des propriétés du modèle. L'objet est constitué de trois régions de propriétés différentes : deux ballons et un tube les reliant. Le ballon de gauche est une région musculaire, le tube une région solide et le ballon de droite une région élastique. Le volume intérieur de l'objet doit rester constant pendant toute la simulation.

2.1.1 Modèle

L'objet a été engendré synthétiquement (génération d'un ensemble de points répartis sur une sphère et calcul de l'enveloppe convexe à l'aide du programme *nnsort*⁴). La partie musculaire et la partie élastique sont similaires (127 éléments), la partie solide centrale comporte 12 éléments. 6 éléments de chacun des deux ballons sont contraints à rester solidaire à la partie centrale (raccordement par contrainte).

⁴ <http://www.iinet.net.au/~watson/software.html>

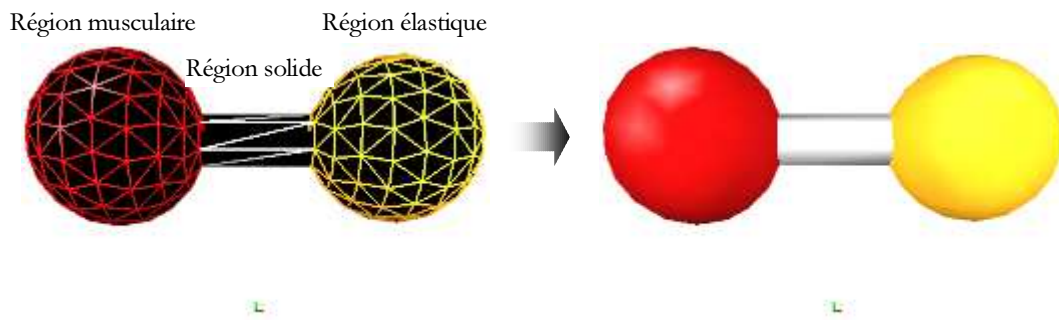


Figure 2. Objet synthétique composé de deux ballons déformables reliés par une barre rigide.

2.1.2 Simulation

La simulation consiste à contracter la région musculaire. On observe alors les conséquences de la résolution de la contrainte de volume constant. Le muscle relâché, l'objet doit reprendre sa forme initiale (les volumes des deux régions élastiques se rééquilibrant automatiquement).

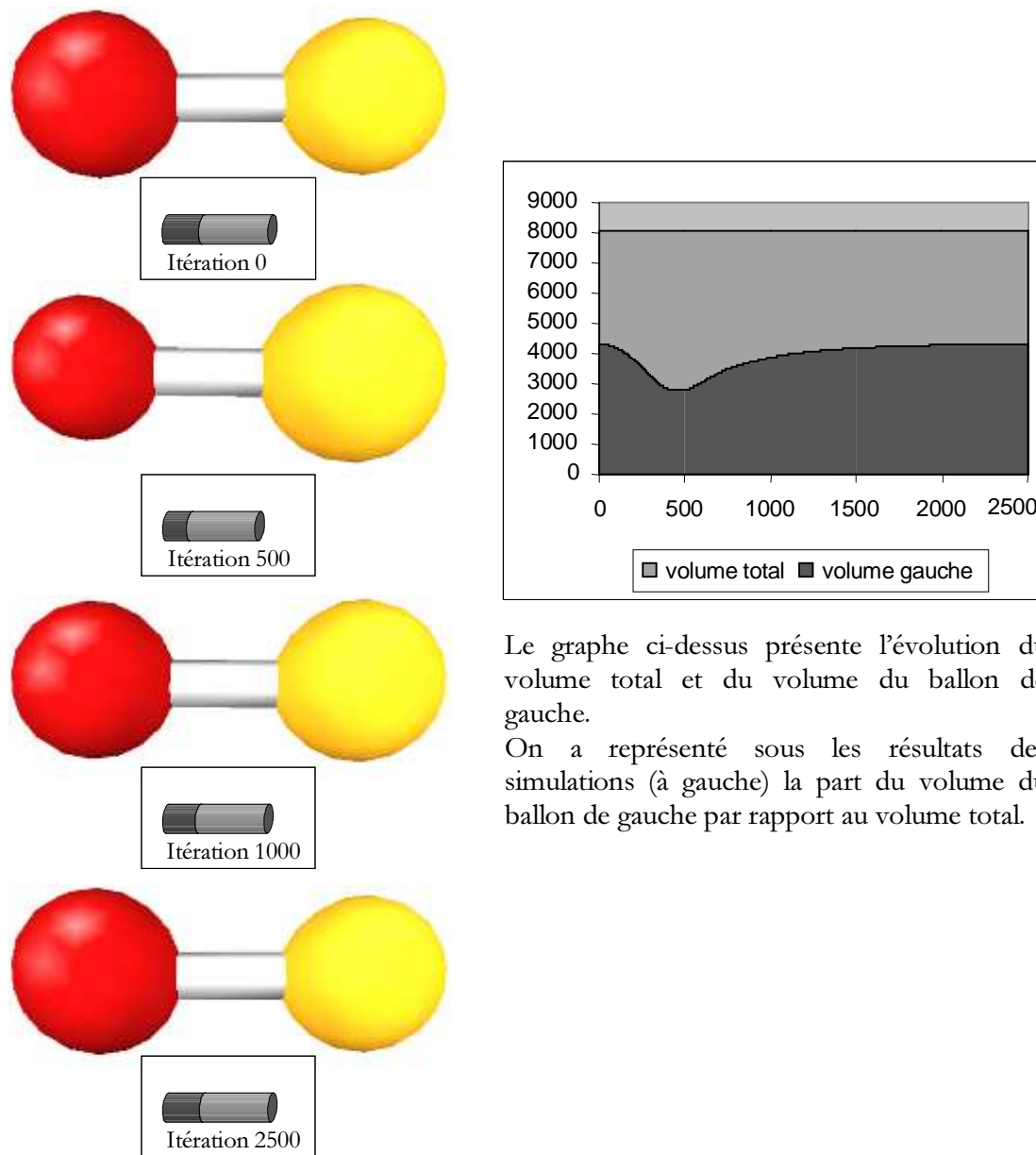


Figure 3. Simulation : la région musculaire est contractée pendant 300 itérations (augmentation constante du poids de la force musculaire), puis relaxation en 300 itérations (diminution constante du poids de la force musculaire). Le temps de calcul (utilisateur) pour une itération est de 0,032s sur une DEC AlphaStation 500/500Mhz.

2.1.3 Conclusion

Le volume est conservé durant toute la simulation (à la précision de la machine près). La région musculaire retrouve sa forme de manière automatique grâce à son élasticité.

On peut noter cependant un problème inhérent au modèle d'élasticité choisi : le retour est plus lent que la contraction (et ce quelle que soit la valeur du paramètre d'élasticité utilisée). Cela provient du fait que le poids de la force élastique reste constant quelle que soit l'amplitude de la déformation. Une solution envisageable est de transformer notre modèle d'élasticité linéaire en faisant varier le poids de la force élastique selon une fonction de l'amplitude de déformation. Pour cela, on peut utiliser notamment les coefficients α , β et γ pour déterminer l'amplitude de la déformation en chaque élément.

Une conséquence importante de la méthode choisie pour résoudre la contrainte de volume constant est la suivante : le transfert de volume est automatique entre les deux régions. Ceci s'explique par le fait que les forces agissant sur les éléments de gauche restent plus importantes dans la première phase que celles agissant sur les éléments de droite. Ainsi, en résolvant la contrainte, et bien que le déplacement dû à la contrainte en chaque élément soit équivalent à gauche comme à droite, c'est la partie droite qui gonfle dans la première phase. Lorsque les forces de contractions sont annulées, les forces élastiques permettent de rétablir l'équilibre entre les deux parties. Ainsi, pour créer un matériau compressible de manière hétérogène on peut utiliser le poids de la force élastique : plus le poids de cette force en un élément est important, moins cet élément sera compressible.

2.2 Bras

Dans ce paragraphe l'exemple utilisé est celui de la contraction d'un muscle en vue d'obtenir un mouvement du squelette. Outre l'étude de la contrainte de raccordement entre région élastique et région solide et entre deux régions solides, nous montrons l'intérêt de la déformation à volume constant non seulement sur le plan de la modélisation (un muscle se contracte à volume constant), mais aussi sur celui de l'animation en infographie (cohérence de la forme induite par la contrainte).

2.2.1 Modèle

Le modèle du squelette provient d'un fichier au format 3D Studio (disponible sur internet⁵) filtré pour être utilisé dans notre modèle.

La partie osseuse comprend l'omoplate (région solide fixée, 289 éléments), l'humérus (région solide fixée, 134 éléments), le cubitus (région solide, 120 éléments), le radius (région solide, 108 éléments) et la main (région solide, 328 éléments). L'omoplate et l'humérus sont fixés et ne bougeront pas pendant la simulation. Tous les éléments du cubitus et du radius ont une masse nulle sauf deux éléments situés aux extrémités des os (dans la direction principale d'inertie). Un élément situé à l'extrémité supérieure de chacun de ces os est fixé (modélisation de l'articulation au coude fixé). Les 328 éléments de la main ont une masse non nulle. La main est reliée au cubitus par 4 liaisons de type région solide/région solide et par 3 liaisons de même type au radius (donc 7 liens de type ressort).

⁵ <http://www.3dcafe.com/anatomy.htm>

Le biceps a été créé synthétiquement à partir d'un ellipsoïde. Il est modélisé par une région musculaire de 80 éléments dont 3 éléments sont liés au cubitus par une contrainte de raccordement entre région solide et région élastique, 4 au radius et 7 à l'humérus.

La Figure 4 montre la modélisation et l'objet 3D obtenu.

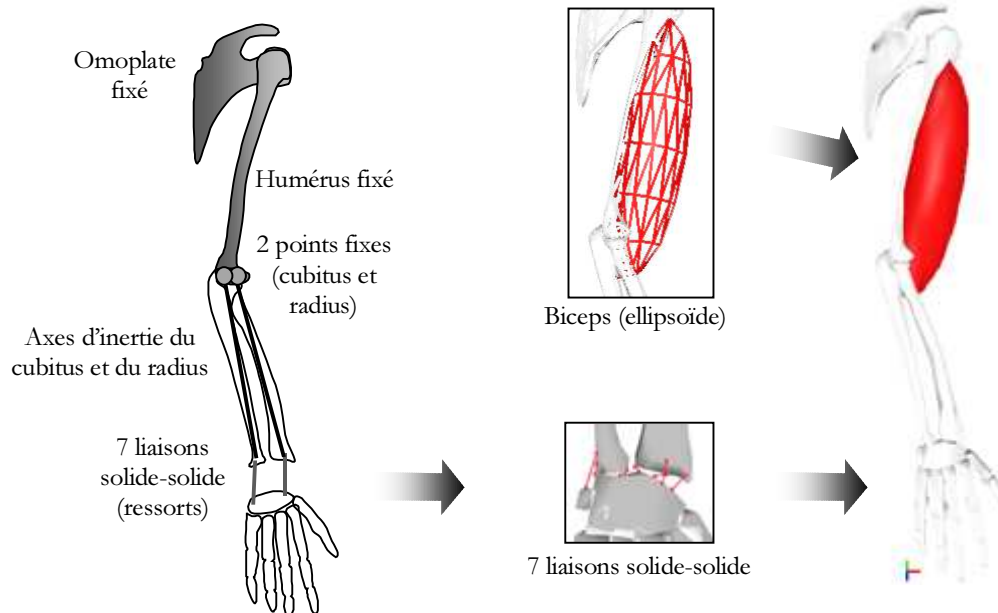
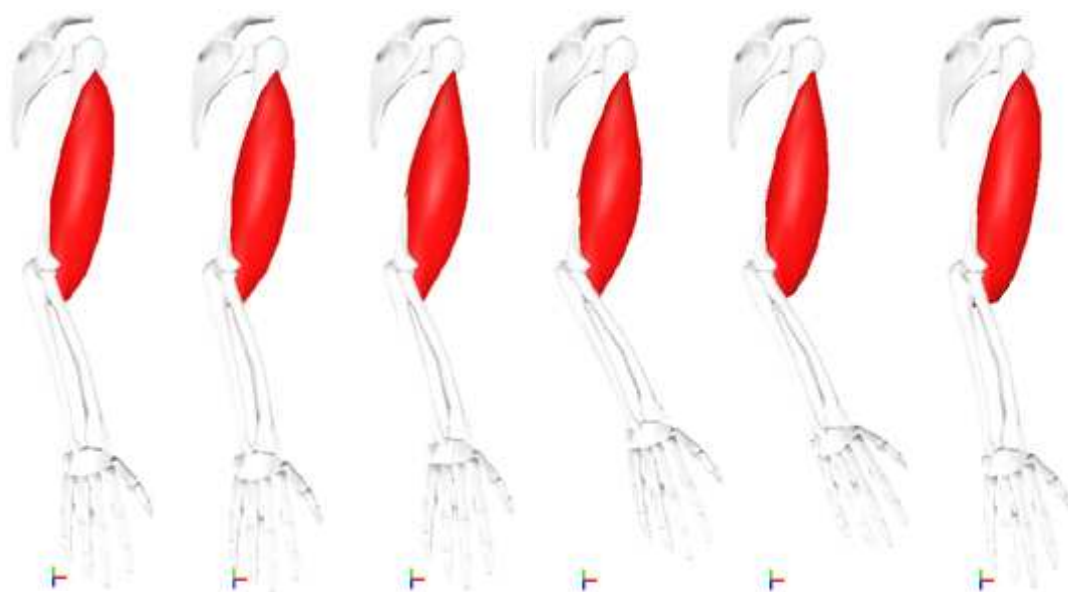


Figure 4. Le modèle du bras. Le cubitus et le radius sont modélisés par deux éléments de masse non nulle aux extrémités de leur axe principal d'inertie.

2.2.2 Simulation

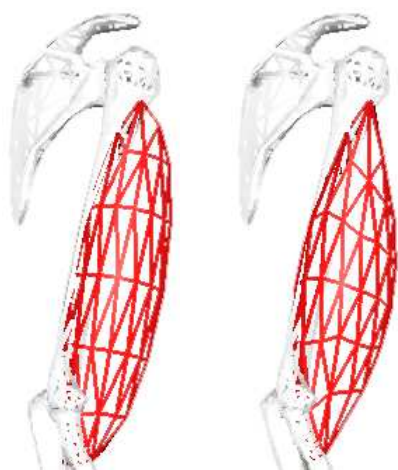
La simulation consiste à laisser le bras sous l'influence de la gravité pendant 500 itérations (l'avant bras s'affaisse). Ensuite on contracte le muscle pendant 1000 itérations, on le laisse contracté pendant encore 1000 itérations, puis on relâche cette contraction petit à petit.

On montre deux simulations. Dans la première la contrainte de volume constant est employée sur le muscle (Figure 5), mais pas dans la seconde (Figure 7).



Itération 0 Itération 750 Itération 1500 Itération 2500 Itération 3500 Itération 5400

Figure 5. Contraction du biceps à volume constant. Itération 1 : état initial, itération 750 : position de l'avant bras la plus basse (250 itération de contraction ont déjà été effectuées), itération 1500 : la contraction est maintenue jusqu'à l'itération 2500. Itération 3500 : la contraction est à nouveau nulle. Itération 5400 : le bras est de retour à la position la plus basse (identique à l'itération 750).



Itération 500

Itération 1500

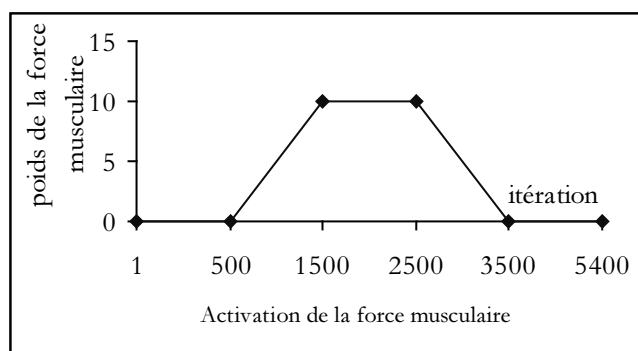


Figure 6. Détail de la contraction du muscle pour les itérations 500 (début de contraction), 1500 (fin de la phase d'accroissement de la fonction de contraction). À droite fonction d'activation au cours des itérations.

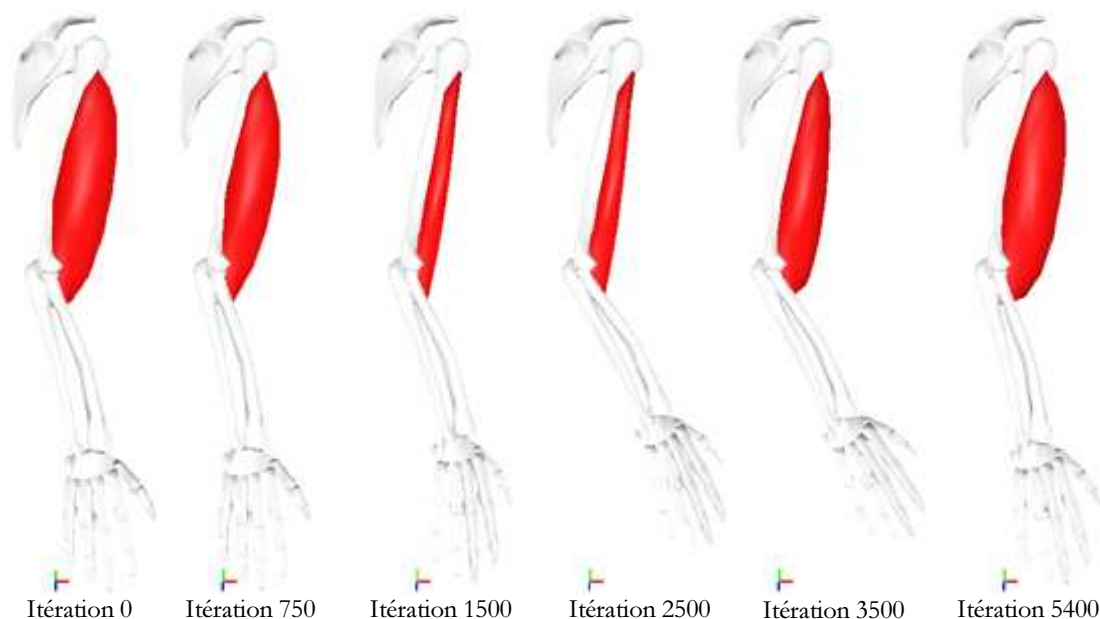


Figure 7. Contraction du biceps sans volume constant (même fonction d'activation de la force musculaire que précédemment).

2.2.3 Conclusion

Les temps de calcul sont ici de 0,025s par itération lorsqu'il n'y a pas de volume constant et de 0,029s par itération lorsque la contrainte est imposée.

Comme précédemment la fonction élastique permet un retour correct à la position initiale.

La fonction d'activation (voir Figure 6) n'a pas de base physiologique. Il est par contre tout à fait possible d'utiliser une fonction d'activation plus proche de la réalité (comme celle utilisé par (Chen, 1992) par exemple).

Un retard de la force de rappel des ressorts de ces liens entraîne un retard du mouvement (la main suit le mouvement du bras avec un léger décalage). Cela est totalement réaliste et modélise des liaisons par les tendons. Si on désire éviter ce genre de problème, par exemple dans le cadre de modélisation de chaînes articulées, il faut faire appel à une résolution de contraintes (celle-ci ne devant pas forcément être une résolution exacte).

La comparaison de la Figure 5 et de la Figure 7 nous fournit une autre motivation pour l'utilisation de la contrainte de volume constant en animation : l'imposition de la contrainte permet le maintien de la forme générale de la région (ici un ellipsoïde) sans changer son action (génération du mouvement du squelette). Elle évite donc d'ajouter d'autres types de liaisons masse-ressort ou d'autres types de force.

2.3 Conception d'objets à volume constant

Nous abordons ici une autre façon d'utiliser le modèle plus lié à la conception ou à la déformation d'objets par un manipulateur. Lors de conception d'objets par manipulation, il

est souvent utile de pouvoir garantir une variation de volume nulle. La contrainte de volume constant, à densité constante, garantit la conservation de la masse. Cette propriété peut être particulièrement utile dans certaines opérations de déformation.

2.3.1 Déformation à volume constant

Dans un premier temps nous n'utiliserons pas le modèle physique développé au cours de cette thèse, mais simplement la résolution de la contrainte de volume constant. On s'appuie sur le fait que la résolution du volume constant par notre méthode déplace les éléments selon leur normale à la surface. Ce qui veut dire que la forme générale avant et après l'application de la contrainte ne change que très peu.

On part d'un objet en forme de banane (disponible sur internet). Le fichier de représentation est filtré pour être utilisé dans notre modèle. L'objet final est composé d'une seule région de 272 éléments (500 facettes). Le volume de cette région est mémorisé, puis on déforme la surface de l'objet à l'aide de l'interface sans tenir compte de la variation de volume. Enfin on affecte à l'objet déformé le volume mémorisé initialement. Ainsi entre l'objet initial et l'objet final le volume est conservé (bien que les phases intermédiaires autorisent une variation de volume). De plus, par le type de résolution employé, l'objet diffère très peu avant et après application de la contrainte.

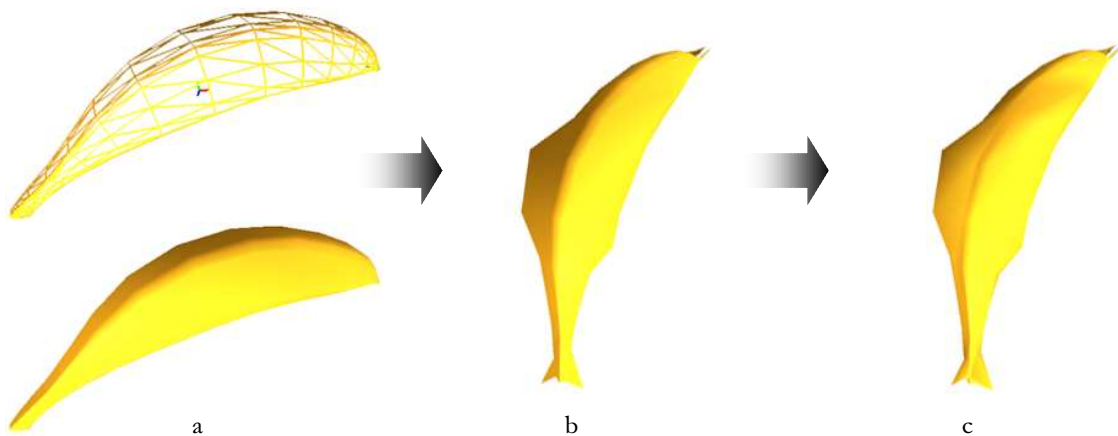


Figure 8. La banane et le « dauphin » ont le même volume. a) l'objet initial $V = 1,805$ l., b) l'objet déformé $V = 1,962$ l., et c) l'objet déformé à qui l'on a affecté le volume de l'objet initial ($V = 1,805$ l.).

2.3.2 Déformation interactive à volume constant

De la même façon on peut se servir de la simulation et des forces élastiques tout en déformant l'objet à volume constant. Ceci est possible car la résolution de la contrainte est rapide (au moins sur des objets de taille moyenne). Dans l'exemple qui suit, on part d'une

sphère de 80 éléments (engendrée mathématiquement). Ces éléments sont regroupés en une région élastique dont on contraint le volume.

On déforme l'objet en déplaçant les éléments à l'aide d'une contrainte (l'élément sélectionné est contraint à suivre les déplacements de la souris).

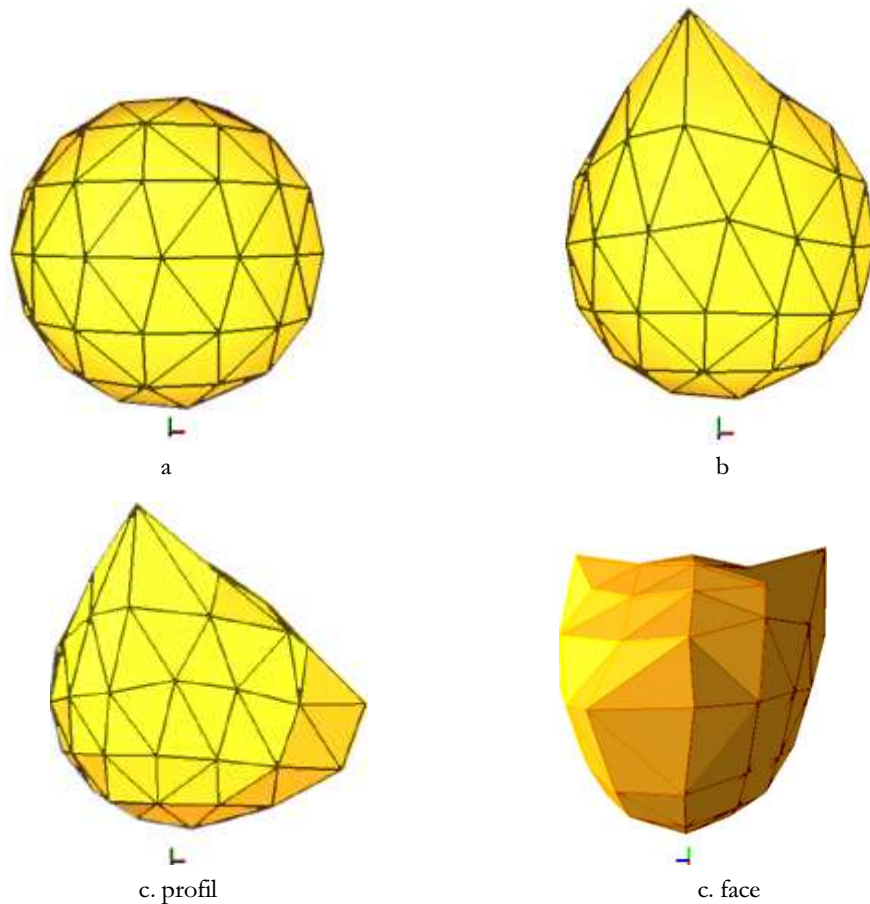


Figure 9. Déformation interactive à volume constant. a) forme initiale. b) en déplaçant un élément vers le haut, le maillage local autour de cet élément suit grâce à la force de mémoire de forme. c) montre la forme finale après des déplacements interactifs de plusieurs autres éléments. L'objet final (une « tête de chat » ?) le même volume que la sphère initiale.

En déplaçant un élément, on génère une force élastique qui entraîne une déformation locale autour de cet élément. La force élastique permet de lisser la déformation (voir Figure 9). Le temps de calcul nécessaire au modèle est de 0,009s par itération ce qui est plus que suffisant pour de la déformation en temps réel.

2.3.3 Conclusion

Nous avons présenté ici deux exemples indirectement liés à notre objectif initial. Pourtant, en utilisant ce même principe, nous sommes capables de modéliser le poumon. On rappelle que le poumon est un élément passif qui adhère toujours aux parois de la cage thoracique (sur un sujet sain). De plus sa déformation est principalement isotropique (égale dans toutes les directions). Comme il est possible de déduire le volume des poumons à partir du volume de la cage thoracique (en soustrayant le volume du médiastin, constant), nous pourrions déformer les poumons en fixant à chaque itération leur nouveau volume.

3. Modèle du tronc

L'objectif initial de cette thèse est la modélisation des mouvements du tronc pendant la respiration. Ce paragraphe montre l'utilisation de notre modèle informatique dans ce but. Après avoir étudié la bibliographie des modèles du tronc en physiologie (paragraphe 3.1), nous présenterons un modèle simplifié du tronc (paragraphe 3.2). Notre démarche se poursuit ensuite par la construction du modèle complet puis par l'étude séparée des différents composants du modèle (paragraphe 3.3 et 3.4). Ce paragraphe se termine par un premier résultat de simulation d'un cycle respiratoire (paragraphe 3.5).

3.1 Bibliographie

Nous présentons ici les principaux modèles de la mécanique du tronc proposés dans la littérature physiologique. (Konno et Mead, 1967) sont les premiers à exploiter le fait que le volume pulmonaire peut être déduit des mesures bi-dimensionnelles du tronc : une mesure de l'extension de la cage thoracique et l'autre de l'extension de l'abdomen. C'est le modèle à deux compartiments. La cinématique du tronc peut alors être décrite par deux variables : la variation de volume de la cage thoracique ΔV_{ct} et la variation de volume de l'abdomen ΔV_a (en fait il s'agit de la variation de volume du compartiment abdominal) :

$$\Delta V = \Delta V_{ct} + \Delta V_a$$

(Primiano, 1982) présente un outil général pour la modélisation du tronc utilisant les balances de pressions.

Le modèle de (Mead et Loring, 1982) inclut des relations anatomiques plus réalistes entre le diaphragme et la cage thoracique. Cependant ce n'est pas un modèle général (il n'est pas aussi complet que le précédent).

(Ben-Haim et Saidel, 1990) proposent une approche phénoménologique des forces, où les modèles d'équations sont aussi dérivés des balances de pression et des relations géométriques entre les compartiments. Les résultats de ces simulations sont consistants quantitativement avec les données disponibles dans la littérature.

(Ward et al., 1992) étendent un modèle antérieur. La cage thoracique est divisée en une partie opposée au poumon et une autre opposée au diaphragme. C'est le premier modèle qui donne deux degrés de liberté à la cage thoracique. De plus ce modèle gère la nature duale du diaphragme.

Tous ces modèles ont été construits en regard des techniques d'acquisition de données existantes (essentiellement des mesures des pressions et des volumes dans différents

compartiments du tronc). Ils ont l'avantage de décrire de manière fonctionnelle le comportement des structures qu'ils simulent. L'anatomie n'est prise en compte que succinctement pour définir des compartiments liés à ces structures.

Aujourd'hui on dispose d'autres systèmes de mesures incluant l'aspect géométrique du tronc au cours de la respiration. Un premier pas dans cet aspect géométrique de la mécanique du tronc a été franchi par (Kaye et al., 1997), voir aussi (Kaye et al., 1995), qui présentent un modèle cinématique et dynamique. Leur modèle est basé à la fois sur la physiologie, à partir des équations de pression/volume définies dans (Primiano, 1982), et sur l'anatomie (segmentation de coupes tomographiques). La cage thoracique, le diaphragme et le médiastin sont modélisés de manière cinématique (leurs déplacements sont interpolés à partir des valeurs de pressions et de volumes avec lesquelles ils sont associés). Chaque poumon est modélisé à la fois cinétiquement (par les équations de pression/volume) et dynamiquement (un noyau de déformation anisotropique et une surface reliée au noyau par des ressorts classiques se déformant isotropiquement). Un environnement 3D complet est ainsi défini et permet l'observation et la modulation des différents paramètres.

Bien que ces derniers travaux puissent être reliés à celui présenté dans cette thèse, on notera cependant que les mouvements du tronc ne sont pas entièrement pris en compte (absence d'abdomen) et que ce sont les équations physiologiques qui pilotent les déformations. Notre approche est quelque peu différente puisque nous désirons un modèle mécanique permettant d'engendrer entièrement les mouvements.

Nous allons présenter maintenant notre modélisation du tronc.

3.2 Modèle simplifié

Nous avons déjà vu au chapitre 1 l'importance de l'incompressibilité de l'abdomen. Cette propriété est essentielle car elle permet à l'abdomen de jouer le rôle de point de levier lors de la contraction du diaphragme. Dans un premier temps nous avons donc cherché à représenter le système composé de l'abdomen, du diaphragme et d'une cage thoracique très simplifiée. Ce modèle a été présenté à la 3^{ème} European Conference of Mathematics applied to Biology and Medicine⁶, Heidelberg (Allemagne), en octobre 1996 (« a physically-based model for modelling the respiratory movements of the trunk », E. Promayon, P. Baconnier, C. Puech), (voir aussi (Promayon et al, 1997)).

3.2.1 Modèle

Pour cela nous avons construit un objet composé de trois régions : une région élastique représentant la paroi abdominale (63 éléments), une région musculaire représentant le diaphragme (17 éléments) et attachée sur une région solide (11 éléments) représentant une paire de côtes fictives. La région élastique et la région musculaire délimitent l'abdomen (de forme ellipsoïde) dont le volume doit rester constant. Certains éléments de l'objet sont fixés dans l'espace afin de représenter les os du bassin et le rachis (au total 12 éléments sont fixés), voir Figure 10.

Lors de la contraction de la région musculaire, et grâce à la propriété d'incompressibilité de l'abdomen, il y a bien élévation de la région solide symbolisant la cage thoracique.

⁶ <http://www.iwr.uni-heidelberg.de/ECMBM96/>

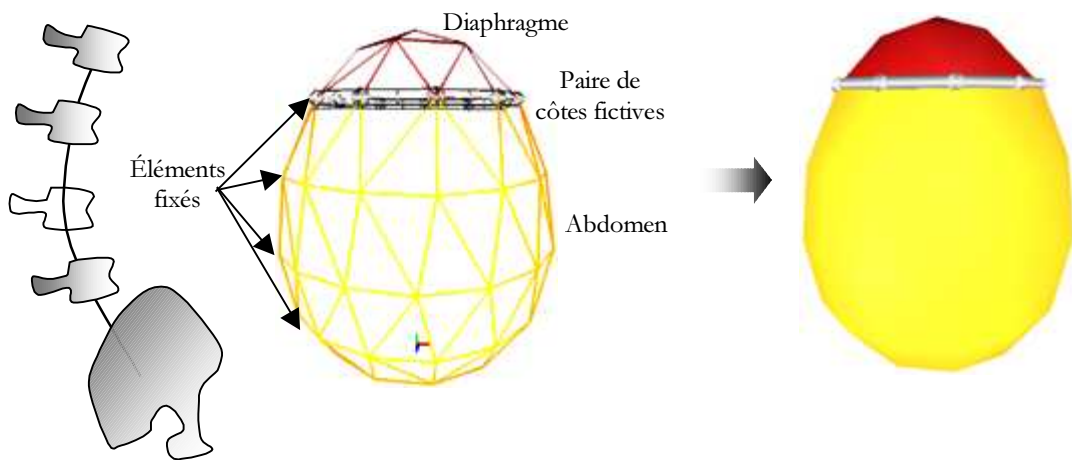


Figure 10. Modèle simplifié de la mécanique du tronc pendant la respiration.

3.2.2 Simulation

La simulation d'un cycle respiratoire consiste à contracter le muscle du diaphragme pendant 50 itérations, à laisser le muscle contracté de la sorte pendant 200 itérations, puis à relaxer le muscle. Ce cycle est représenté à la Figure 11.

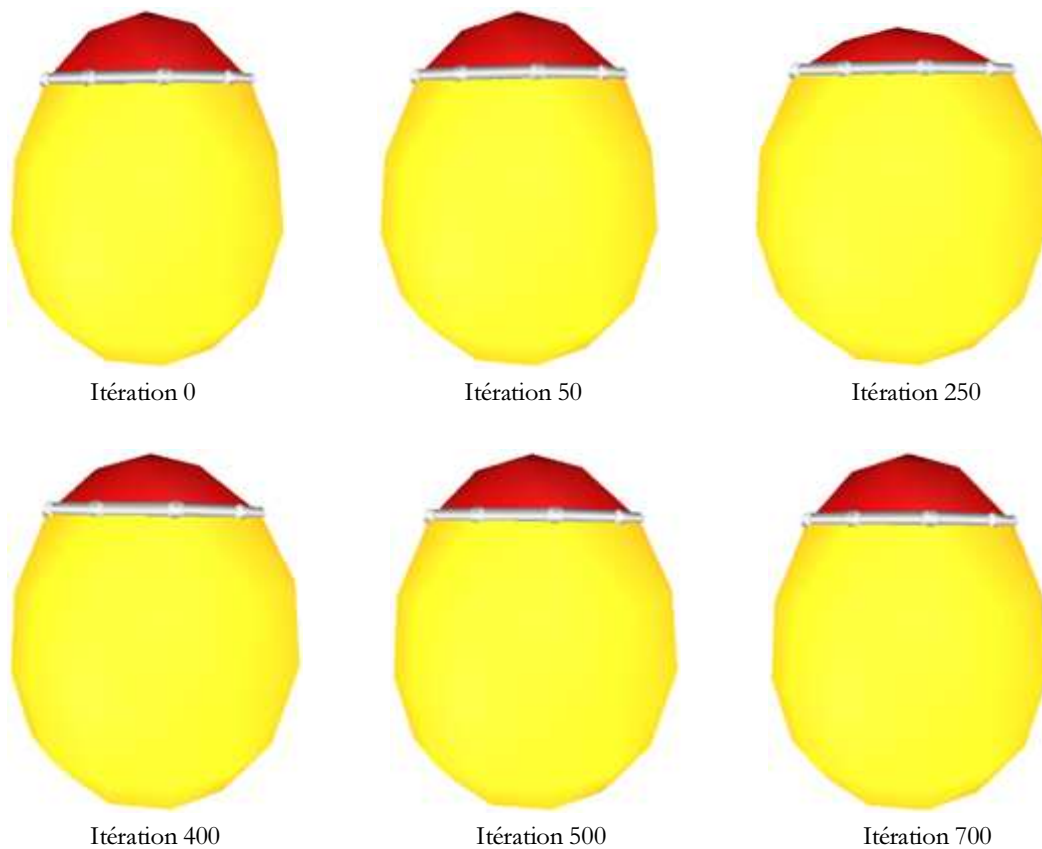


Figure 11. Cycle respiratoire sur le modèle simplifié. De l'itération 0 à l'itération 50 : contraction musculaire, même état de contraction jusqu'à l'itération 250. Ensuite la force musculaire décroît et l'objet retourne à sa forme initiale.

3.2.3 Conclusion

Le temps de calcul est de 0,008 secondes par itération (0,005 secondes lorsque l'on enlève la contrainte sur le volume. La même simulation mais avec un abdomen non contraint en volume ne présente qu'un très faible soulèvement de la paire de côtes. Grâce à ce modèle simplifié nous avons pu vérifier le fonctionnement global de la mécanique du tronc.

3.3 Construction du modèle

Nous allons maintenant montrer un modèle plus complet du tronc. Pour le construire nous sommes partis de données anatomiques et nous avons utilisé trois sources différentes. Pour la partie osseuse de la cage thoracique, nous nous sommes servis de la même base de données que celle du bras. Chacune des côtes (ainsi que le sternum) a ensuite été simplifiée en disposant des éléments selon la courbe principale de la côte.

L'abdomen a été obtenu à partir de données d'un autre projet du laboratoire. Un tronc de mannequin miniature a été numérisé par balayage laser selon différents angles de vues. Les points 3D, ainsi acquis puis mis en correspondance, ont ensuite été utilisés pour reconstruire la surface par la méthode des δ -snakes (voir (Bittar et al, 1993)). À partir la surface 3D du tronc, nous avons éliminé grâce à notre programme les données correspondant à la partie supérieure de la cage thoracique pour ne garder que la partie abdominale, puis nous avons régularisé la disposition des éléments.

L'observation des coupes IRM du projet Visible Human⁷, nous a permis de concevoir (par manipulation d'une surface) la région musculaire correspondant au diaphragme.

Le modèle du tronc (voir Figure 12) dont nous disposons est donc une approximation puisqu'il ne provient pas entièrement d'un même sujet type.

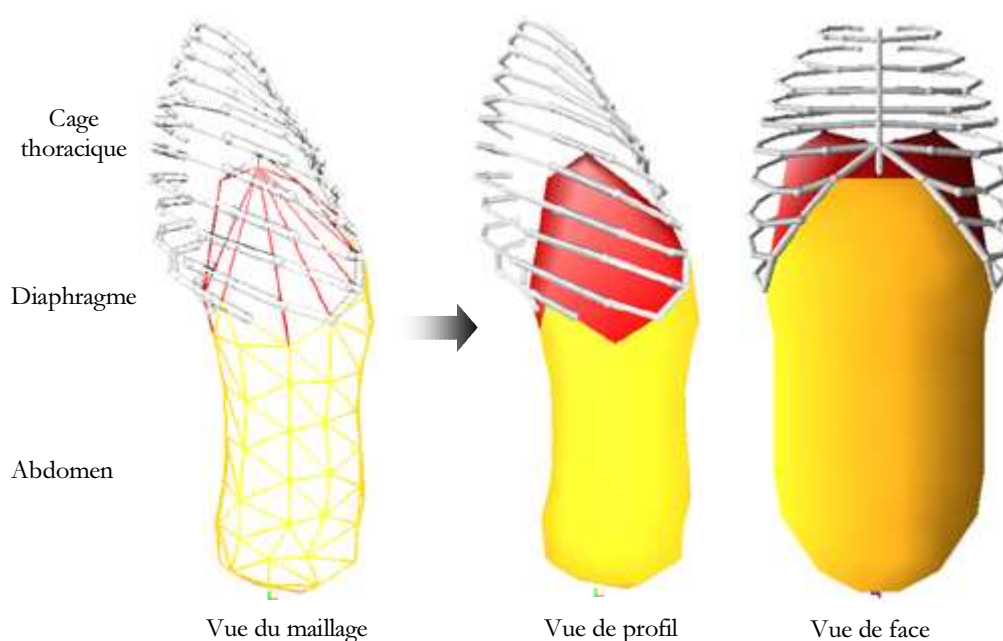


Figure 12. Modèle du tronc construit pour la simulation.

Nous allons d'abord étudier chaque structure séparément (paragraphe 3.4) puis nous montrerons une simulation d'un cycle respiratoire (paragraphe 3.5).

3.4 Étude séparée des structures composant le modèle complet

L'étude séparée des 3 structures (cage thoracique, parois abdominales et diaphragme) nous permettra de vérifier leurs propriétés spécifiques et de calibrer ces propriétés avant l'assemblage en un seul objet.

⁷ <http://www.npac.syr.edu/projects/vishuman/VisibleHuman.html>

3.4.1 La cage thoracique

Pour obtenir un modèle simplifié et géométriquement correct, on a extrait les courbes des côtes à partir de leurs représentations sous forme de facettes. On a ainsi obtenu 20 régions solides de 7 à 9 éléments représentant les 10 premières paires de côtes, 4 régions solides de 4 ou 5 éléments représentant les 2 paires de côtes flottantes et une région solide de 16 éléments représentant le sternum et la partie antérieure des côtes (voir Figure 13).

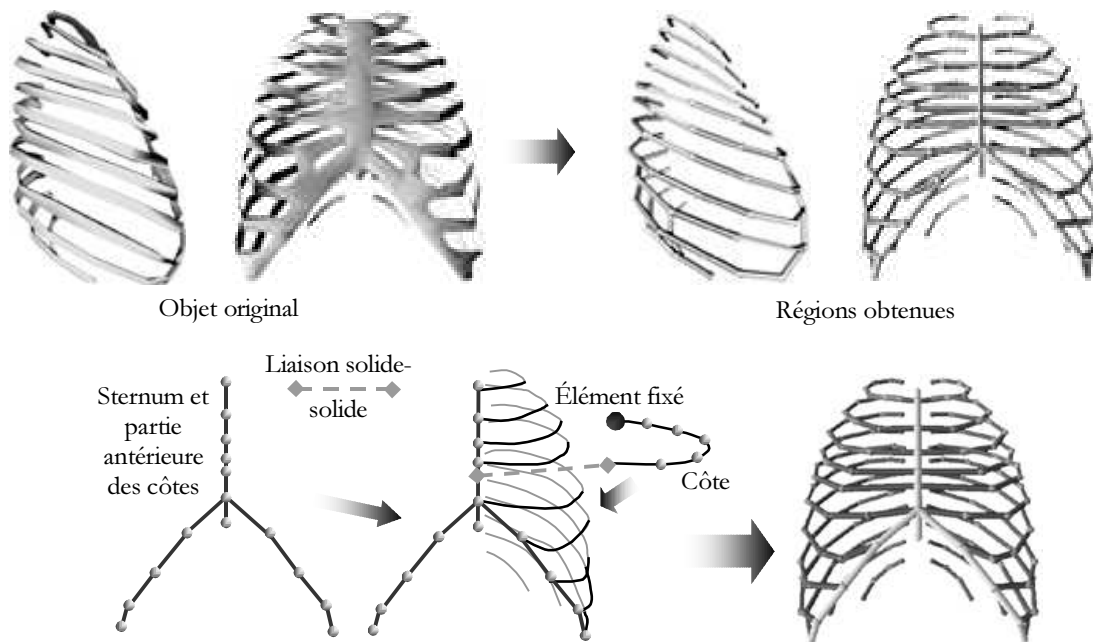


Figure 13. Obtention des 25 régions solides modélisant la cage thoracique (une région solide par côte et une région centrale). L'élément dorsal de chaque côte est fixé, l'élément antérieur est positionné au même endroit que son homologue du sternum et est relié à celui-ci par une liaison entre régions solides (type ressort).

Pour étudier le comportement et calibrer les liaisons au niveau du sternum, une force de gravité (vers le bas) est appliquée à la région centrale. Une fois cette force enlevée la cage thoracique doit retourner à peu près à sa forme et position de départ du fait de son élasticité. Cette simulation est montrée à la Figure 14.

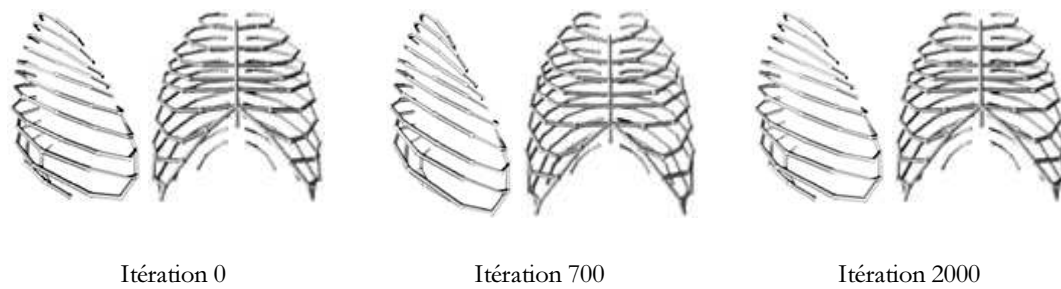


Figure 14. Une force de gravité appliquée à la région centrale de la cage thoracique entraîne l'ensemble des côtes vers le bas (itération 1 à 700). Une fois la force annulée, la cage thoracique revient peu à peu à sa position de départ par élasticité.

Le même type de modélisation est employé par (Closkey et al, 1992), qui, pour étudier des déformations pathologiques de la cage thoracique (dans le cas de scolioses par exemple), considère un modèle d'éléments rigides articulés à l'aide de ressorts. Cependant cette modélisation ne fait pas appel à la dynamique (seuls les comportements statiques et des positions d'équilibre sont simulés).

3.4.2 L'ensemble composé par l'abdomen et diaphragme

Dans ce paragraphe on étudie le comportement de l'ensemble délimitant l'abdomen : les parois abdominales et le diaphragme. C'est ici que la propriété de volume constant sera utilisée.

Les parois abdominales sont constituées par la simplification d'un modèle de la surface d'un tronc. Le diaphragme est obtenu par manipulation d'une surface (voir Figure 15).

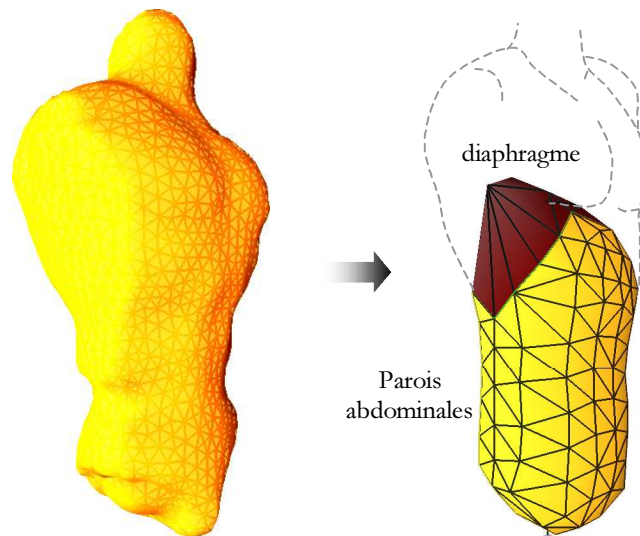


Figure 15. Ensemble délimitant l'abdomen : parois abdominales (98 éléments) et diaphragme (32 éléments).

50 éléments de la région élastique représentant les parois abdominales sont fixés (représentant le rachis et les os du bassin, que l'on considère comme fixe dans notre modèle). Seuls les deux éléments dorsaux du diaphragme sont fixés.

Comme dans le modèle simplifié, nous allons contracter la région musculaire du diaphragme et observer le comportement de l'abdomen. Toujours afin de calibrer les paramètres, nous effectuons cette contraction sur l'ensemble abdomen-diaphragme sans le relier à la cage thoracique (Figure 16).

Le muscle du diaphragme est d'abord contracté, il en résulte une déformation des parois abdominales (afin de respecter la contrainte de volume constant). Le muscle est ensuite relâché et l'objet reprend sa forme initiale. On peut déjà noter qu'ici encore le retour à la forme initiale est plus lent que la déformation.

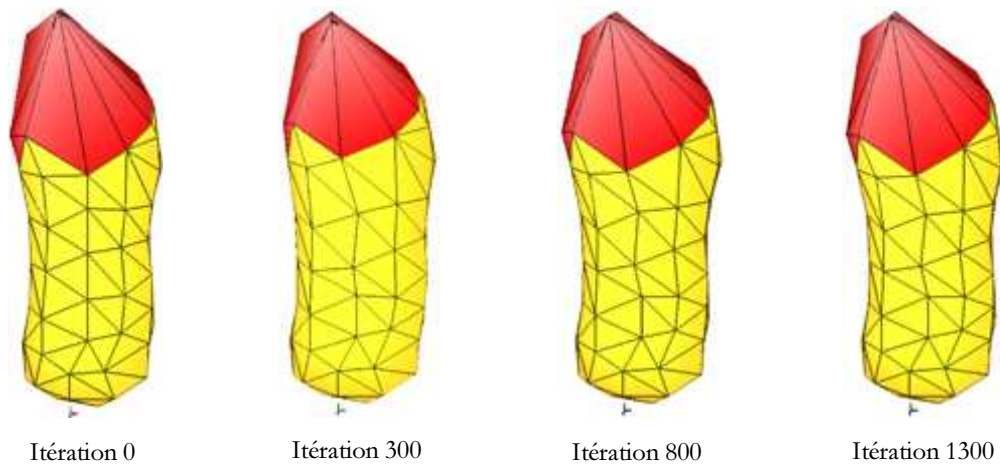


Figure 16. Une phase de respiration sur l'ensemble abdomen-diaphragme (contraction de l'itération 0 à l'itération 300, puis relaxation de la force musculaire). Le temps de calcul est de 0,013s par itération.

Chaque composant du modèle complet a été étudié systématiquement, nous avons calibré grossièrement les paramètres du modèle global. Maintenant nous allons rassembler les différentes structures et utiliser les mêmes paramètres sur le modèle global.

3.5 Simulation

Le modèle complet représente 314 éléments répartis en 27 régions (25 régions solides, une région élastique et une région musculaire), 76 éléments sont fixés.

Pour lier la cage thoracique au diaphragme on utilise la contrainte de raccordement entre les éléments de la partie costale du diaphragme. Cette contrainte porte sur 8 éléments seulement.

On reprend les paramètres de contraction trouvés pour la structure abdomen-diaphragme. Le temps de calcul est de 0,017 s par itération. Le résultat est présenté à la Figure 17.

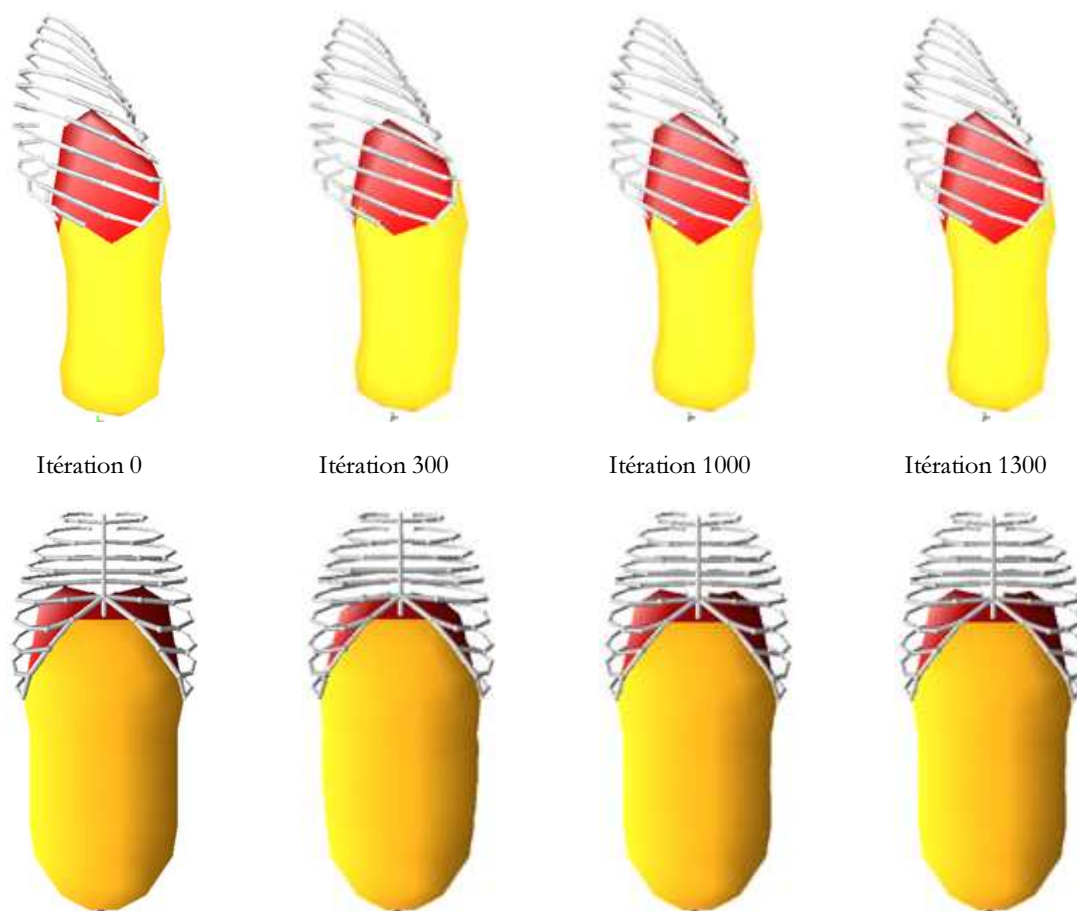


Figure 17. Simulation d'un cycle respiratoire. Contraction du muscle de l'itération 0 à l'itération 300, puis relaxation. L'abaissement du diaphragme et le gonflement de l'abdomen est bien représenté. L'élévation des côtes est aussi présente comme dans la réalité, mais et très faible, ce qui correspond aussi à un cycle respiratoire de repos.

3.5.1 Conclusion

On a vu tout au long de ces simulations que le temps de calcul par itération pour les modèles de taille moyenne que l'on utilise est très court (de l'ordre du centième de seconde sur la station de travail utilisée, une DEC Alphastation 500 cadencée à 500 Mhz). Pour les simulations de la Figure 17, par exemple le calcul d'une itération (en temps utilisateur et non pas temps CPU) nécessite 0,017 secondes. Cependant le temps d'affichage des données à l'écran demande lui 0,087 secondes (dont 0,07 secondes pour l'affichage 3D par OpenGL), et ce bien que l'accélérateur graphique de la machine utilisée soit très puissant. On s'aperçoit donc que, même si la parallélisation de l'algorithme est possible (tous les calculs, sauf une

partie de celui de la contrainte de volume constant qui nécessite au total 0,001 secondes, peuvent se faire au niveau des éléments), elle ne permettrait pas un gain de temps significatif. Le modèle du tronc met en œuvre des interactions assez complexes et le premier modèle que l'on présente ici ne saurait décrire un large spectre de comportements. Cependant, et bien que nous n'ayons pas encore validé quantitativement le modèle, une validation qualitative préalable (géométriquement correcte) a été fournie.

Nous allons maintenant, en guise de perspective, exposer brièvement les techniques et les méthodes à mettre en œuvre pour obtenir une simulation plus exacte de la réalité.

4. Comparaison et validation

Le problème posé par toute modélisation est celui de la validation. La question fondamentale est de savoir si le modèle reproduit (ou peut reproduire) correctement la réalité. Nous avons vu qu'il était possible de reproduire des comportements similaires à la réalité à l'aide du modèle générateur développé dans cette thèse. Nous cherchons maintenant à déterminer dans quelle mesure les comportements fournis par ce modèle sont conformes aux comportements réels.

Pour influencer sur le comportement obtenu en sortie du modèle de déformation du tronc, nous pouvons soit manipuler les paramètres de l'objet (masse et poids de forces), soit sa géométrie. Pour simplifier, nous considérons dans ce paragraphe l'objet tronc défini plus haut (cage thoracique, abdomen et diaphragme) comme une constante. Le problème revient donc à identifier les paramètres afin d'obtenir un comportement correct.

Dans la littérature infographique ce problème reste encore très ouvert. La plupart des recherches se situent plus sur le plan de la paramétrisation à but d'optimisation que sur celui de l'identification : les paramètres conservés sont ceux qui permettent un mouvement plus rapide ou plus fluide. On peut citer (Grzeszczuk et Terzopoulos, 1995) qui optimise la trajectoire d'objets modélisés par un modèle générateur grâce à l'apprentissage.

L'identification de paramètres peut être vue sous l'angle de la minimisation d'erreur (on cherche à minimiser l'erreur commise par la simulation par rapport à la réalité). Dans ce cas on peut utiliser les méthodes classiques de minimisation :

1. Recherche empirique : elle peut être exhaustive (génération d'un treillis dans l'espace des paramètres et recherche du meilleur vecteur) ou basée sur des méthodes itératives (il faut alors que la continuité soit garantie et ceci reste à prouver dans de tels modèles).
2. Descente de gradient : il faut alors expliciter la fonction du modèle et sa dérivée (ce qui ne paraît pas réalisable). La continuité est là encore nécessaire.
3. Réseau de neurones : on peut citer (Szilas, 1996), où un réseau de neurones calqué sur l'objet 3D équivalent de Cordis-Anima (voir chapitre 2) est utilisé dans une phase d'apprentissage afin de reproduire un comportement réel. L'inconvénient est qu'il faut développer un réseau pour chaque objet à paramétrer. L'intervention du manipulateur et son interaction sur le réseau donne cependant de bons résultats.
4. Algorithmes de recuit simulé,
5. Algorithmes génétiques : on peut citer (Garat, 1996), voir aussi (Joukhadar, 1997), qui utilise un algorithme génétique pour identifier les paramètres du modèle Robot- Φ développé par (Joukhadar, 1997). Le manque de données réelles n'a pas permis aux

auteurs une expérimentation poussée, les premiers résultats sur des cas analytiques semblent prometteurs.

6. Minimisation par des méthodes de moindres carrés : ces méthodes restent les plus connues et les plus utilisées en physiologie et en ingénierie pour la santé. Elles sont faciles à implémenter et de nombreuses optimisations ont été développées dans la littérature.

En plus des problèmes classiques rencontrés dans d'autres domaines (par exemple celui des minima locaux), l'identification de paramètres par minimisation de l'erreur est confrontée à d'autres difficultés que sont la détermination des bornes des paramètres ou la multiplicité des solutions pour un même résultat. Les bornes des paramètres sont données soit par les limites de convergences du modèle utilisé soit par la limite de plausibilité des solutions par rapport à la réalité. Cependant, dans le cas de modèle générateur issus de l'informatique, la question est posée de la correspondance des paramètres du modèle et des paramètres physiques réels.

Pour déterminer la complexité du problème d'identification de notre modèle du tronc nous avons déterminé le jeu de paramètres le caractérisant le plus simplement (voir paragraphe 4.1). Nous examinons dans le paragraphe 4.2 les données réelles disponibles utilisables dans cette phase d'identification.

4.1 Paramètres à mettre en correspondance

Les paramètres intrinsèques du modèle du tronc dépendent de chacune des régions de l'objet représentant le tronc. On doit cependant distinguer ou regrouper des sous-ensembles internes de paramètres en corrélation avec les indépendances fonctionnelles internes aux régions.

On regroupe par exemple l'ensemble des côtes de gauche et celles de droites pour former deux héli-thorax. L'ensemble des régions solides représentant les côtes gauches auront donc les mêmes paramètres (indépendants de ceux des côtes de droite). Par contre on doit distinguer 4 sous-parties dans le diaphragme (une division antérieure-postérieure en distinguant la partie costale de la partie crurale, et une division latérale). La Figure 18 explicite le jeu de paramètres nécessaire à la distinction fonctionnelle.

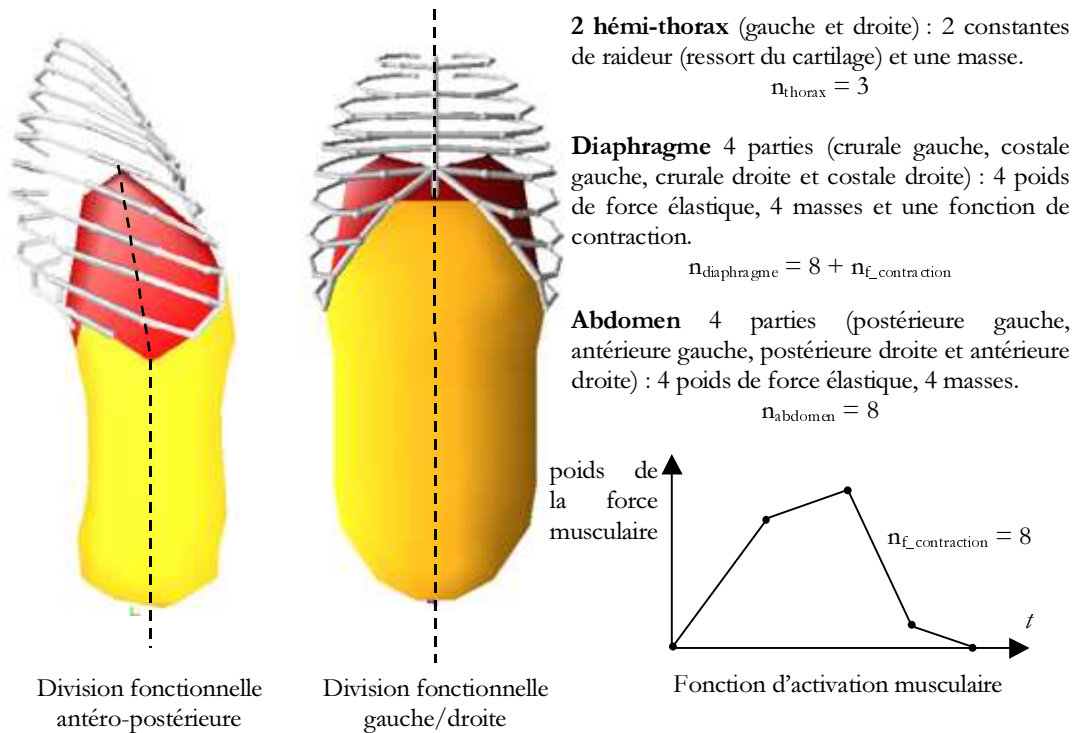


Figure 18. Distinction fonctionnelle des paramètres et nombre de paramètres résultants. Le paramètre de masse est appliqué à tous les éléments de la région concernés.

Le nombre minimal de paramètres est donc de :

$$n_{\text{param}} = n_{\text{thorax}} + n_{\text{diaphragme}} + n_{\text{abdomen}} = 19 + n_{f_{\text{contraction}}}$$

où $n_{f_{\text{contraction}}}$ est le nombre de paramètres à déterminer pour modéliser la fonction de contraction (voir Figure 18) : 3×2 coordonnées et 2 abscisses, obtenus par simplification de données physiologique. On obtient donc :

$$n_{\text{param}} = 27.$$

Ce qui reste très raisonnable pour une première approximation.

Par contre, si on cherche à identifier des pathologies du diaphragme (hémidiaphragme ou autre), il est nécessaire de séparer l'abdomen en huit parties distinctes (division supérieure/inférieure) et d'utiliser une fonction d'activation musculaire pour chacune des quatre parties du diaphragme. On arrive alors à $n_{\text{param}} = 55$.

On pose des hypothèses suivantes : le pas de simulation est constant et la structure géométrique (même fichier) de l'objet 3D est invariante.

Les limites de validité de la phase d'identification des paramètres d'un modèle (issus de techniques informatiques ou mathématiques) sont directement liées à l'expérimentation. Seules des données expérimentales directement mesurées sur le phénomène étudié peuvent

servir à l'identification des paramètres. La validité du modèle est, elle, déduite de sa capacité à restaurer les paramètres corrects à partir de données expérimentales bruitées.

Les données expérimentales disponibles ont été décrites dans le Tableau 1 du chapitre 1. Il s'agit des mesures de variations de pressions, de volumes, et de positions. Notre modélisation étant basée sur la géométrie du système, c'est principalement sur la géométrie et la cinématique du système que doit se baser l'identification.

Nous proposons ici un algorithme d'identification simple (voir principe Figure 19) :

7. Identification des paramètres de la fonction d'activation musculaire à partir des courbes de débit de volume
8. Utilisation d'un algorithme de minimisation d'erreur de position pour l'identification des autres paramètres : à partir des données dynamiques sur les éléments (position, vitesse et accélération), on peut retrouver les forces appliquées à tout moment et ainsi, après lissage, déterminer des nouvelles valeurs de paramètres minimisant l'erreur.

Nous verrons au paragraphe 4.2 quelles données sont actuellement disponibles afin de procéder à cette deuxième étape.

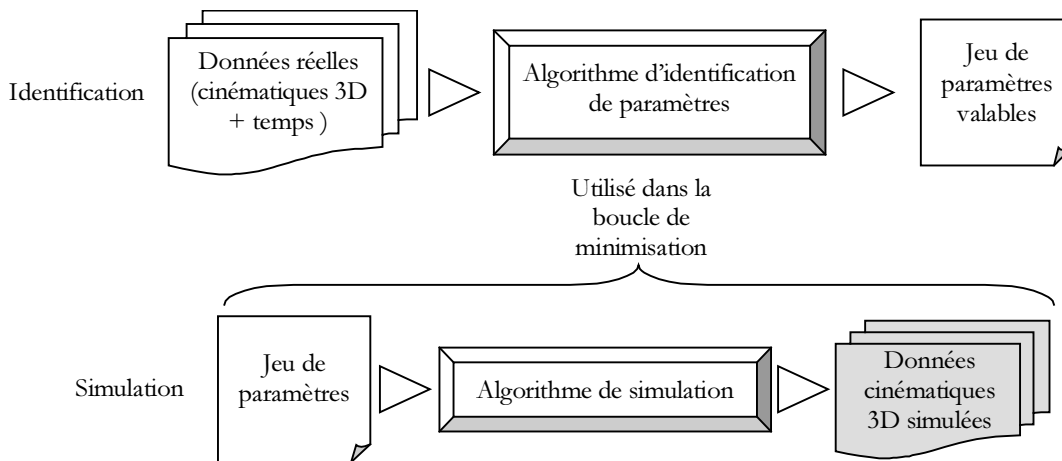


Figure 19. Principe de l'identification des paramètres.

Au-delà même de l'identification des paramètres, une autre étude doit être menée pour déterminer la validité du modèle et sa sensibilité au bruit. Cette étape est importante puisqu'elle permettra de déterminer dans quelle mesure on peut trouver une correspondance entre les paramètres du modèle et les paramètres réels. La démarche (voir Figure 20) consiste dans un premier temps et une fois un jeu de paramètres valide (comprenant le fichier de description de l'objet) déterminé, à brouter ce jeu de paramètres et à produire une sortie graphique à l'aide de la simulation (cette sortie est incorrecte puisqu'elle est issue de paramètres incorrects). Dans un deuxième temps, cette sortie graphique incorrecte sera utilisée afin de recalibrer le système (identification des paramètres à partir de la sortie graphique incorrecte). La robustesse du modèle est mesurée par sa capacité à retrouver alors le jeu de paramètres valide initial.

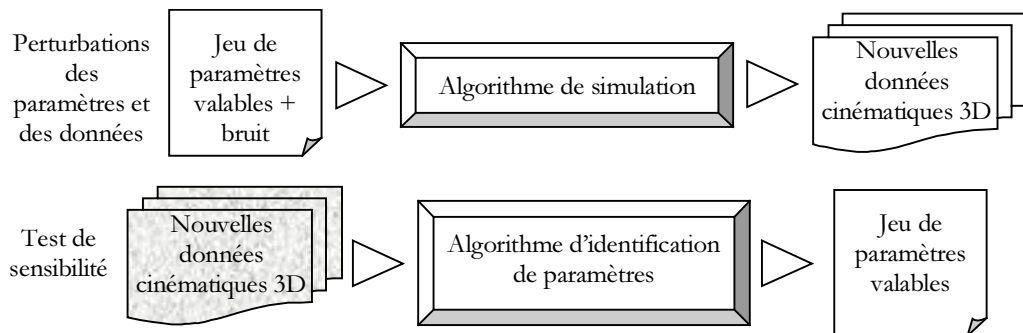


Figure 20. Validité des paramètres et sensibilité du modèle.

L'algorithme d'identification est établi dans notre cas sur une base géométrique. Nous allons maintenant voir ce que l'algorithme d'identification peut utiliser comme données tridimensionnelles.

4.2 Données cinématiques : mouvements du tronc

L'analyse des mouvements du tronc a d'abord été faite par l'intermédiaire de jauges de contrainte (fixées à l'intérieur d'un gilet) ou de pléthysmographes à variation d'inductance. Ces mesures sont partielles puisqu'elles ne se font qu'en deux ou trois endroits du tronc (par exemple le thorax et l'abdomen). Elles sont cependant directement utilisables par les modèles physiologiques classiques (voir paragraphe 3.1). De telles approches posent de nombreux problèmes : calibrage des mesures à l'aide d'un spiromètre, dépendance forte avec la manière de respirer, imposition de la posture du sujet. D'autres systèmes mettent en jeu des mesures optiques mais nécessitent une segmentation manuelle, ce qui est une source d'erreur importante. D'autres systèmes encore utilisent des tomographes 3D, le problème étant ici la lenteur des acquisitions et l'aspect invasif de la technique (rayons X).

Une équipe de Milan a mis au point le système optique 3D ELITE permettant la détection et la mesure de marqueurs passifs 3D positionnés sur le corps (Ferrigno et al, 1990). Ce système, n'induisant pas les problèmes cités plus haut, a été adapté pour la mesure des mouvements du tronc, voir (Ferrigno et al, 1994).

4.2.1 Mesure des mouvements par système optique 3D

À l'aide de 50 marqueurs positionnés le long du tronc (grille de 5×5 marqueurs sur la face antérieure et postérieure du tronc) : voir (Carnevali et al, 1996) (et Figure 21), on peut mesurer les volumes de 16 compartiments. Quatre caméras sont disposées autour du sujet pour mesurer le déplacement des marqueurs puis sont calibrées. La fréquence des acquisitions vidéo est de 25 Hz. Ce système a été validé par de nombreuses mesures sur sujets sains dans différentes postures, en comparant notamment les données avec des mesures spirométriques. L'erreur moyenne en position allongée (seule la moitié antérieure du tronc est visible) est inférieure à 9% de la variation de volume (elle n'est que de 2% quand le sujet est debout ou assis).

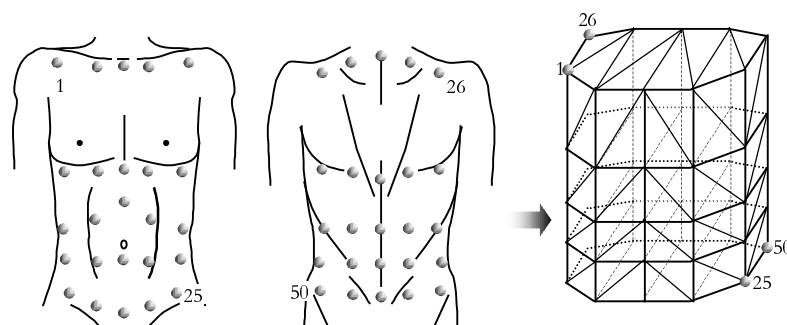


Figure 21. Positionnement de marqueurs (à gauche) et calcul des variations des volumes sur les 16 compartiments correspondants (à droite).

Après un premier contact avec l'équipe du centre de bio-ingénierie du Politecnico de Milan, nous avons concrétisé notre collaboration par l'intermédiaire du projet européen BREATH (programme BIOMED) qui a démarré en novembre 1997, pour une période de 3 ans. Notre participation s'inscrit dans le cadre de la modélisation du phénomène et devrait permettre de coupler la mesure et le modèle pour mettre au point une technique d'analyse (et à terme de surveillance). Dans une première phase cette collaboration nous fournira une très grande base de données cinématiques 3D du tronc et nous permettra d'identifier les paramètres de notre modèle.

4.2.2 Interfacer notre modèle

Comme le système ELITE mesure seulement la surface du tronc, notre modèle doit encore être modifié pour pouvoir utiliser directement ces données. La couche épidermique sera modélisée simplement par une enveloppe de type surface implicite, comme dans (Volino et al, 1996) ou (Wilhelms, 1997), qui suivra directement les mouvements des structures dans un premier temps. Par la suite on pourra utiliser une région ayant des propriétés élastiques particulières pour modéliser différentes physiologies.

4.3 Conclusion

De meilleures mesures entraîneront à l'avenir le développement de modèles meilleurs. Ces modèles, en retour nous permettront de choisir les mesures les plus judicieuses et de prédire le comportement des variables non mesurées. [...] La relation entre mesure et prédiction doit être complémentaire ; les avancées de l'une à la fois dépendent et stimulent les avancées de l'autre.

S.H. Loring, E.N. Bruce, (Loring et Bruce, 1986)

L'équipe de radiologie centrale du CHU de Grenoble, intéressée par le modèle présenté ici, met actuellement en place un projet d'étude des mouvements du diaphragme à l'aide d'un nouvel appareil IRM à acquisition rapide. Notre collaboration se situera à deux niveaux : d'un

côté les données ainsi collectées permettront d'améliorer la modélisation du muscle dans notre modèle, et d'un autre côté, le modèle que nous avons développé permettra d'aider à l'obtention de données (par interpolation des positions du diaphragme et de certains éléments topologiques).

Les techniques modernes de mesure permettent aujourd'hui d'envisager l'approfondissement des connaissances de la mécanique du tronc. Cependant ces techniques d'acquisition nécessitent des nouveaux modèles qu'elles contribuent à développer en fournissant des nouvelles données.

Dans le cas de modèles tels que ceux développés au cours de cette thèse, l'étape suivante, primordiale, est l'identification et la validation des paramètres du modèle. Cependant, la question reste ouverte de savoir s'il y a correspondance fonctionnelle entre les paramètres intrinsèques du modèle et la réalité. Si cette correspondance existe, les modèles générateurs de l'infographie sont promis à un bel avenir en physiologie.

Conclusion

Il n'existe pas aujourd'hui d'outil permettant de modéliser la mécanique ventilatoire de manière globale, tout en étant précise. Cette thèse est un premier pas vers l'élaboration d'une modélisation et d'une simulation comportementale des mouvements du tronc pendant la respiration. Dans cette optique j'ai mis en place une approche méthodique de modélisation. Ce document est entièrement basé sur cette approche. Je conclus ici sur les contributions apportées par cette étude. Pour finir, différentes perspectives de ce travail, débuté il y a trois ans, sont explorées.

1. Contribution

À l'issue de ce travail, on peut lister les apports généraux :

- La démarche de modélisation expliquée en introduction et illustrée par cette thèse définit une méthodologie générale pour la modélisation informatique.
- L'interaction de régions hétérogènes au sein d'un même objet permet d'obtenir des comportements complexes. Le modèle développé au cours de cette thèse permet de manipuler de tels objets.

On peut considérer de manière plus précise les contributions originales suivantes :

- Développement d'une approche et d'un modèle « orienté-objet » définissant des classes d'objets en fonction des propriétés qu'ils doivent simuler.
- Mise au point d'une nouvelle représentation de la forme locale permettant de définir une fonction d'élasticité plus stable que la méthode classique masse-ressort.
- Présentation d'un modèle de muscle simple mais permettant le contrôle de l'activation musculaire et des directions de contractions ainsi que la transmission des forces musculaires aux structures solides ou élastiques liées.
- Développement d'une nouvelle méthode directe de résolution de contraintes (locales ou globales) calculant directement les déplacements nécessaires à la vérification exacte des contraintes.
- Application de cette méthode pour la modélisation de la propriété d'incompressibilité.
- Généralisation de cette méthode pour dégager ses principes afin de l'utiliser dans d'autres modèles pour simuler des objets déformables incompressibles.

- Construction de modèles du tronc (modèle simple et modèle plus réaliste).

Le modèle informatique, sur lequel se base ces apports, a été validé qualitativement grâce à des simulations sur des exemples.

On trouvera en annexes de ce manuscrit des documents plus « pratiques » (simulation d'objets rigides, utilisation des quaternions, définition des normales et description d'objets dans notre modèle).

Ce travail a donné lieu à deux publications dans des conférences internationales (Promayon et al, 1996), (Promayon et al, 1997) et à deux communications dans des colloques internationaux (Conference on Dynamical Systems in Biology and Medicine, Veszprém, juillet 96 et 3rd European Conference on Mathematic applied to Biology and Medicine, Heidelberg, octobre 1996).

De plus deux collaborations à but clinique ont été démarrées. La première collaboration, en partenariat avec le service de radiologie centrale du CHU de Grenoble, a pour but de coupler le modèle du tronc à un appareil d'Imagerie par Résonance Magnétique à acquisition rapide et d'étudier ainsi les mouvements du diaphragme. La deuxième collaboration se fera dans le cadre du projet européen BREATH (programme BIOMED 2). Elle a pour but l'utilisation du modèle du tronc dans un système optique de mesure des mouvements du tronc au cours de la respiration. Ce projet européen est actuellement en phase de démarrage.

2. Perspectives

Ces deux collaborations peuvent certainement être considérées comme un excellent cadre pour les perspectives de ce travail. Cependant, il faut d'abord examiner les différents problèmes posés par le modèle informatique développé ici (problèmes déjà soulevés dans la dernière partie de ce manuscrit). En perspective on peut déjà donner des pistes pour la résolution de ces problèmes :

- L'élasticité définie dans notre modèle provoque un retour lent à la forme de départ. Ce retour doit être amélioré (notamment pour pouvoir simuler des phases respiratoires rapides). Pour cela on peut définir une fonction hyper-élastique (fonction du potentiel de déformation subi) par l'intermédiaire des trois paramètres de la mémoire de forme α_0 , β_0 , et γ_0 .
- La fonction d'activation musculaire n'a pour l'instant jamais été utilisée sur des bases physiologiques. Pour palier à cet inconvénient il faut étudier les formes d'activation pour les muscles concernés et valider leurs réponses dynamiques.

L'amélioration de notre modèle passe par les éléments suivants :

- Le problème des frottements (et des collisions) n'a pu être abordé dans cette thèse. Or il semble nécessaire de l'étudier pour parfaire la modélisation d'un système biologique comme celui de la respiration où les interactions inter-structurelles sont importantes.
- La plasticité n'a pas encore pu être étudiée non plus. On peut considérer que la mise à jour interactive des paramètres de forme à partir d'une combinaison des paramètres de

l'étape initiale et de l'instant t de la déformation, fournit une réponse simple à cette modélisation.

- La méthode de résolution que nous avons présentée est justifiée dans le cadre de petits déplacements. Il faut étudier la limite au-delà de laquelle elle fournit une réponse erronée à la résolution de la contrainte (cas où la déformation entre deux itérations est très importante).
- La résolution de contraintes locales (fixation des éléments) et globales (volume constant) fonctionne parfaitement. Mais, nous n'avons jusqu'à présent pas pu tester le comportement de notre méthode de résolution lorsque plusieurs contraintes globales sont en jeu.
- La surface épidermique n'est pour l'instant pas présente dans le modèle. Des approches hiérarchiques (multi-couches) sont utilisées en infographie pour créer des modèles d'animaux (humain compris). L'application des techniques ainsi développées permettrait de fournir une réponse à ce manque (inversement les modèles hiérarchiques pourraient se servir de notre modèle comme couche motrice interne).

Les potentialités contenues dans les paramètres de formes définis au chapitre 3 sont nombreuses et méritent d'être étudiées. En effet, les paramètres α_i , β_i et γ_i donnent une information directe sur la géométrie locale. On peut imaginer utiliser cette caractéristique dans des algorithmes de mise en correspondance, de reconnaissance de formes, de métamorphose (« morphing ») ou encore de simplification d'objets.

Outre la modélisation simple de la plasticité, ces paramètres pourraient servir dans des algorithmes de mise en correspondance. Ils pourraient aussi être utilisés dans la simplification de maillages (en les utilisant comme une base de décomposition de la forme locale).

En guise d'ouverture voici trois éléments de réflexion sur l'apport du modèle présenté dans cette thèse :

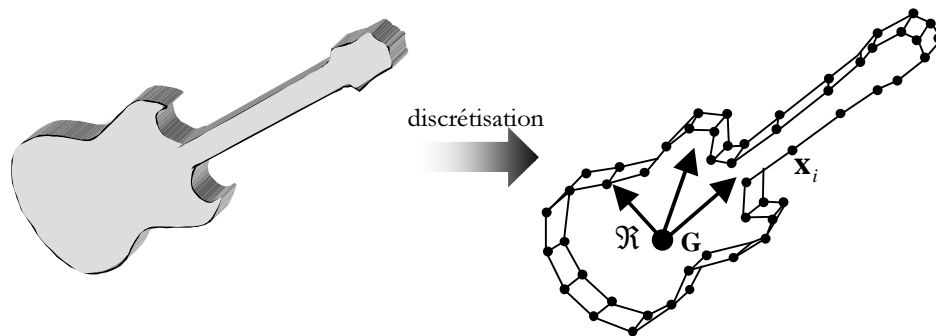
- Le modèle développé dans cette thèse fournit un outil intéressant à de nombreux titres. Outre l'exploration fonctionnelle de la respiration (ou d'autres systèmes biologiques) qu'il permet d'envisager, il pourrait servir de manière plus technique dans des procédés d'acquisition pour le pilotage des appareils.
- Les modèles générateurs sont aujourd'hui, grâce au développement de la puissance de calculs des ordinateurs, très étudiés dans le domaine de l'interface homme machine ou dans celui de la communication au travers des mondes virtuels. L'acteur virtuel prend place peu à peu dans notre environnement et le modèle de la respiration présenté dans cette thèse peut en être un élément.
- L'informatique est certainement l'outil d'aide à la modélisation le plus puissant disponible à l'homme. Fruits de son utilisation, de nouveaux domaines de recherche commencent aujourd'hui à émerger. L'étude de la mécanique cellulaire (objets comportant des régions élastiques et quasi-solides souvent à volume constant) par l'intermédiaire de modèles générateurs est un parfait exemple de défi qui s'ouvre devant nous.

Annexe 1

Implémentation de la dynamique d'un objet solide (système de points matériels)

À partir des formules données au chapitre 2 on va expliciter directement l'algorithme de traitement d'un objet rigide, voir aussi (Barzel et Barr, 1988).

L'objet rigide est composé de n points matériels. Les points matériels sont indexés par leur numéro. Pour un point matériel i on désigne sa position par $\mathbf{x}_i = (x_i, y_i, z_i)$, sa masse par m_i et la somme des forces qui lui sont appliquées par \mathbf{f}_i . Ce seront les trois entrées de notre algorithme.



L'algorithme de cette annexe a été utilisé pour l'implémentation des régions solides de notre système de simulation physique (classe des régions solides).

Pour simplifier les étapes de calcul, on définit un repère local \mathcal{R} attaché à l'objet dont l'origine est le centre des masses \mathbf{x}_G et dont les axes sont les axes principaux d'inertie de l'objet (mais on peut choisir n'importe quelle autre orientation initiale).

Par la suite, quand on exprimera les coordonnées dans le repère local, on mettra \mathcal{R} en indice. Ainsi, on a $\mathbf{x}_G^{\mathcal{R}} = (0 \ 0 \ 0) = \mathbf{0}$.

Les algorithmes sont donnés en pseudo-code C++, on dispose des classes `Vector` et `Matrix` (dans notre implémentation nous avons utilisé directement la librairie d'algèbre linéaire ALP développée au sein du laboratoire par E. Bainville).

1. Données de l'objet

Pour caractériser l'objet on aura besoin de :

```

Int n;           // nombre de points matériels
double m;       // masse du solide
Vector xg;     // centre des masses
Vector vg;     // vitesse linéaire de G
Vector wg;     // vitesse angulaire en G
Vector awg;    // accélération angulaire
Matrix R;      // matrice définissant l'orientation
Quat q;       // quaternion de l'orientation
Quat aq;      // accélération du quaternion

Matrix I $\mathfrak{R}$ ; // matrice d'inertie dans  $\mathfrak{R}$ 
Matrix Ii $\mathfrak{R}$ ; // inverse de I $\mathfrak{R}$ 
Vector mi[n]; // masse de chaque point
Vector xi[n]; // position de chaque point / repère global
Vector vi[n]; // vitesse de chaque point / repère global
Vector xi $\mathfrak{R}$ [n]; // position de chaque point dans  $\mathfrak{R}$ 

```

En plus des opérations spécifiques sur les vecteurs et les matrices, on dispose dans les classes `Vector` et `Matrix` des mêmes opérateurs que pour les réels (addition, multiplication...).

$\mathbf{v}(i)$ est la $i^{\text{ème}}$ composante du vecteur du vecteur \mathbf{v} ,

$\mathbf{m}(i, j)$ est l'élément de la ligne i et de la colonne j de la matrice \mathbf{m} .

$\mathbf{0}$ dénote soit le vecteur dont toutes les composantes sont nulles, soit la matrice dont tout les éléments sont nuls.

2. Initialisations

Certains calculs ne se feront qu'une fois et seront valables tant que :

- Aucune masse m_i n'est changée,
- Aucun point matériel n'est ajouté ou enlevé à notre objet.

Il s'agit du calcul de la masse m de l'objet, du centre des masses \mathbf{x}_G , de la matrice d'inertie dans le repère \mathfrak{R} et de son inverse (qui sont diagonales) et des coordonnées des points matériels \mathbf{x}_i dans le repère \mathfrak{R} (désignées par $\mathbf{x}_i^{\mathfrak{R}}$).

2.1 Calcul du centre des masses

On applique simplement les formules du calcul du centre des masses pour un système composé de n points matériels.

```

m = 0; g = 0;
Pour i de 1 à n
    g += mi[i]*xi[i];
    m += mi[i];
g /= m;

```

2.2 Calcul de la matrice d'inertie et de la matrice et du quaternion d'orientation

On utilise la formule discrète donnée au chapitre 2 pour calculer la matrice d'inertie dans le repère global. Cependant, cette matrice d'inertie sera différente dès que l'objet aura effectué un mouvement. La matrice d'inertie nécessitant beaucoup de calculs, on la calcule par rapport au centre des masses. Ainsi, elle sera indépendante du repère global, et restera constante pendant toute la simulation. À partir de la formule suivante (propriété des matrices de rotations) :

$$\mathbf{I} = \mathbf{R}\mathbf{I}^{\mathfrak{R}}\mathbf{R}^{-1},$$

on va pouvoir trouver la matrice $\mathbf{I}^{\mathfrak{R}}$ car elle est diagonale dans le repère local. En effet, il suffit de calculer la matrice \mathbf{I} et de chercher la matrice \mathbf{R} qui fait passer de $\mathbf{I}^{\mathfrak{R}}$ (diagonale) à \mathbf{I} . On utilise pour cela la décomposition en valeur propre, pour l'algorithme voir (Lascaux et Théodor, 1994).

Au cours de la simulation, à tout moment, on peut obtenir \mathbf{I} à partir de $\mathbf{I}^{\mathfrak{R}}$. De la même façon on aura besoin de l'inverse de $\mathbf{I}^{\mathfrak{R}}$, on va la stocker aussi.

```

double A, B, C, D, E, F;
Matrix I;
Vector eigenVal;
A = B = C = D = E = F = 0;
I = 0;
// calcul des coefficients dans le repère principal
Pour i de 1 à n
    A += mi[i]*(xi[i](2)*xi[i](2)+xi[i](3)*xi[i](3));
    B += mi[i]*(xi[i](1)*xi[i](1)+xi[i](3)*xi[i](3));
    C += mi[i]*(xi[i](1)*xi[i](1)+xi[i](2)*xi[i](2));
    D += mi[i]*xi[i](2)*xi[i](3);
    E += mi[i]*xi[i](3)*xi[i](1);
    F += mi[i]*xi[i](1)*xi[i](3);
// translation en G
A += m*(g(2)*g(2) + g(3)*g(3));
B += m*(g(3)*g(3) + g(1)*g(1));
C += m*(g(1)*g(1) + g(2)*g(2));
D -= m*g(2)*g(3);

```

```

E -= m*g(3)*g(1);
F -= m*g(1)*g(2);
// Remplissage de la matrice I
I(1,1) = A; I(1,2) = -F; I(1,3) = -E;
I(2,1) = -F; I(2,2) = B; I(2,3) = -D;
I(3,1) = -E; I(3,2) = -D; I(3,3) = C;
// Calcul de la matrice d'inertie dans  $\mathfrak{R}$  et de  $\mathbf{R}$ 
SolveEigenProblem(I,eigenVal,R);
// Remplissage des matrices d'inertie
Pour i de 1 à 3
    I $\mathfrak{R}$ (i,i) = eigenVal(i);
    Ii $\mathfrak{R}$ (i,i) = 1/eigenVal(i);
Normalize( R ); // normalise les vecteurs colonnes.
MatrixToQuaternion(R,q);

```

2.3 Calcul des coordonnées des points dans le repère \mathfrak{R}

\mathbf{R} permet de passer du repère local au repère global. Comme \mathbf{R} est une matrice de rotation, son inverse est sa transposée : $\mathbf{R}^t = \mathbf{R}^{-1}$. On utilisera la matrice d'orientation pour tout ce qui est multiplication d'un vecteur car cela est plus économique que d'utiliser le quaternion (voir chapitre 2).

On conserve les coordonnées des points dans le repère local, puisque toutes les translations et rotations seront rapportées sur le centre des masses. À partir de la nouvelle position du centre des masses, on déduira les positions de tous les points.

Vector gxi;

Pour i de 1 à n

```
xi $\mathfrak{R}$ [i] = Rt* (xi[i]-g);
```

2.4 Initialisation des autres variables

Si l'objet est immobile, et qu'on le considère isolé au départ de la simulation (aucune force ne lui est appliquée), on doit initialiser à zéro :

- la vitesse linéaire du barycentre,
- la vitesse angulaire du mobile, et
- l'accélération angulaire du mobile.

3. Simulation d'une itération

On a besoin des paramètres suivants :

- l'ensemble des données sur l'objet à l'instant t (voir paragraphe 1),
- les forces appliquées en chacun des points de l'objet à l'instant t :
Vector fi[n]; // Forces appliquées en chaque point,

- le pas de temps :

```
double dt; // pas de temps
```

Le paragraphe suivant donne l'algorithme qui permet de calculer les nouvelles valeurs de l'accélération linéaire et angulaire à l'instant $t + dt$. Le paragraphe suivant explicitera deux schémas d'intégration pour passer de l'instant t à l'instant $t + dt$.

3.1 Calcul des nouvelles grandeurs (à l'instant $t + dt$)

Il faut d'abord calculer les valeurs des deux éléments du torseur dynamique.

\times dénote le produit vectoriel.

Vector s; // Somme des quantités d'accélération

Vector mo; // Somme des moments dynamiques

s = 0; mo = 0;

Pour i de 1 à n

s += fi[i];

mo += (xi[i]-g)×fi[i];

s /= m;

Le principe de calcul de l'algorithme suivant découle de la relation entre la quantité de mouvement angulaire \mathbf{L} et le moment au centre d'inertie \mathbf{M} :

$$\mathbf{M} = \frac{d\mathbf{L}}{dt} = \frac{d\mathbf{I}\mathbf{w}}{dt} = \frac{d\mathbf{I}}{dt} \dot{\mathbf{w}},$$

$$\begin{aligned} \mathbf{M} &= \frac{d\mathbf{I}}{dt} \mathbf{w} + \mathbf{I} \dot{\mathbf{w}} \\ &= -\mathbf{I}\mathbf{w} \times \mathbf{w} + \mathbf{I} \dot{\mathbf{w}} \end{aligned}$$

Ceci est exprimé dans le repère global (même la matrice d'inertie). \mathbf{M} est connu, \mathbf{w} est connue. On peut calculer \mathbf{I} et en déduire $\dot{\mathbf{w}}$:

$$\dot{\mathbf{w}} = \mathbf{I}^{-1} (\mathbf{M} + \mathbf{I}\mathbf{w} \times \mathbf{w})$$

Comme on l'a vu, le calcul de la matrice d'inertie est assez lourd en opérations. On va donc utiliser la matrice d'inertie définie dans le repère local qui est diagonale.

De plus on a :

$$\dot{\mathbf{q}} = \frac{1}{2} (0, \mathbf{w}) * \mathbf{q},$$

où \mathbf{q} est le quaternion représentant l'orientation de l'objet, $(0, \mathbf{w})$ le quaternion obtenu en plongeant \mathbf{w} dans l'espace des quaternions (partie réelle nulle), et $*$ l'opérateur de multiplication de deux quaternions (voir annexe 2 sur l'utilisation des quaternions).

Les étapes initiales seront de transformer la matrice $\mathbf{I}^{\mathfrak{R}}$ et $\mathbf{I}^{\mathfrak{R}}$ dans le repère global à l'aide de la matrice d'orientation :

$$\mathbf{I} = \mathbf{R} \mathbf{I}^{\mathfrak{R}} \mathbf{R}^{-1} \text{ et}$$

$$\mathbf{I}^{-1} = \mathbf{R} \mathbf{I}^{\mathfrak{R}-1} \mathbf{R}^{-1}$$

Ainsi on obtient l'algorithme suivant :

```

Matrix I; //  $\mathbf{I}^{\mathfrak{R}}$  dans le repère global
Matrix Ii; //  $\mathbf{Ii}^{\mathfrak{R}}$  dans le repère global
// Calcul de la matrice  $\mathbf{I}$  et  $\mathbf{Ii}$ 
I = R* Iℝ *Rt
Ii = R* Iiℝ * Rt
// Calcul de l'accélération de la vitesse angulaire
awg = Ii*(M+ ((I*wg) X wg))
aq = 0.5*QuaternionMul(w,q);

```

3.2 Intégration, calcul de la nouvelle position

On a maintenant l'ensemble des grandeurs nécessaires au calcul du déplacement entre l'instant t et l'instant $t + dt$. Les deux schémas d'intégration les plus répandus sont le schéma d'Euler et le schéma de Newton-Cotes (plus précis lorsque les forces sont constantes pendant dt).

3.2.1 Schéma d'Euler

$$\begin{aligned}\dot{\mathbf{x}}_G^{t+dt} &= \dot{\mathbf{x}}_G^t + \ddot{\mathbf{x}}_G^{t+dt} \cdot dt \\ \mathbf{x}_G^{t+dt} &= \mathbf{x}_G^t + \dot{\mathbf{x}}_G^{t+dt} \cdot dt \\ \mathbf{w}^{t+dt} &= \mathbf{w}^t + \dot{\mathbf{w}}^{t+dt} \cdot dt \\ \mathbf{q}^{t+dt} &= \mathbf{q}^t + \left(0, \mathbf{w}^{t+dt}\right) * \mathbf{q}^t \cdot dt\end{aligned}$$

3.2.2 Schéma de Newton-Cotes

$$\begin{aligned}\dot{\mathbf{x}}_G^{t+dt} &= \dot{\mathbf{x}}_G^t + \ddot{\mathbf{x}}_G^{t+dt} \cdot dt \\ \mathbf{x}_G^{t+dt} &= \mathbf{x}_G^t + \frac{1}{2} \left(\dot{\mathbf{x}}_G^{t+dt} + \dot{\mathbf{x}}_G^t\right) \cdot dt \\ \mathbf{w}^{t+dt} &= \mathbf{w}^t + \dot{\mathbf{w}}^{t+dt} \cdot dt \\ \mathbf{q}^{t+dt} &= \mathbf{q}^t + \left(0, \frac{1}{2} \left(\mathbf{w}^{t+dt} + \mathbf{w}^t\right)\right) * \mathbf{q}^t \cdot dt\end{aligned}$$

Les algorithmes ne sont pas donnés car trop proches des formules ci-dessus.

3.2.3 Nouvelle position et nouvelle vitesse

À partir de la nouvelle position du centre des masses et de la nouvelle orientation, on va calculer la position de l'ensemble des points. Pour cela on utilise la matrice de rotation (plus économique dans la multiplication de vecteurs), elle permet de passer du repère local au repère global.


```
// Calcul de la matrice de rotation
QuaternionToMatrix(q,R);
// Nouvelle position et vitesse
Pour i de 1 à n
```

```
    xi[i] = xg + R*xig[i];
```

```
    vi[i] = vg + wg X R*xig[i];
```

Il est important de calculer la vitesse correcte car certaines forces utilisent la vitesse comme paramètre (par exemple la viscosité du milieu).

Annexe 2

Utilisation des quaternions

Nous ne présentons pas ici une étude théorique sur les quaternions ou les rotations mais un ensemble d'algorithmes d'utilisation courante pour la manipulation des quaternions. On se référera pour une étude plus approfondie à (Shoemake, 1985) et (Bainville, 1996).

1. Passage de la matrice de rotation au quaternion

Le quaternion est représenté sous la forme d'un vecteur de dimension 4.

On note $\mathbf{q} = (q_1, q_2, q_3, q_4)$ avec $q_1^2 + q_2^2 + q_3^2 + q_4^2 = 1$.

La matrice de rotation peut s'exprimer en fonction des composantes du quaternion par :

$$\mathbf{R} = \begin{pmatrix} 1 - 2q_3^2 - 2q_4^2 & 2q_2q_3 + 2q_1q_4 & 2q_2q_4 - 2q_1q_3 \\ 2q_2q_3 - 2q_1q_4 & 1 - 2q_2^2 - 2q_4^2 & 2q_3q_4 + 2q_1q_2 \\ 2q_2q_4 + 2q_1q_3 & 2q_3q_4 - 2q_1q_2 & 1 - 2q_2^2 - 2q_3^2 \end{pmatrix}$$

C'est à partir de cette formule qu'on va pouvoir trouver le quaternion.

MatrixToQuaternion(Matrix \mathbf{R} , Quaternion \mathbf{q})

```
double    u[4];
```

```
double    max, val;
```

```
int       imax, i;
```

```
u[0] = 1 +R(1,1) +R(2,2) +R(3,3); // 4 - 4(q22 + q32 + q42) = 4q12
```

```
u[1] = 1 +R(1,1) -R(2,2) -R(3,3); // 4q22
```

```
u[2] = 1 -R(1,1) +R(2,2) -R(3,3); // 4q32
```

```
u[3] = 1 -R(1,1) -R(2,2) +R(3,3); // 4q42
```

```
// Recherche de la plus grande composante
```

```
max = |u[0]|;
```

```
imax = 0;
```

```
Pour i de 1 à 3
```

```
    Si |u[i]| > max alors
```

```
        max = |u[i]|;
```

```
        imax = i;
```

```
// calcul de la plus grande composante de q
```

```
q(imax+1) = sqrt(u[imax]*0.25); //  $q_i = \sqrt{4q_i^2 \cdot \frac{1}{4}}$ 
```

```
val = 4 * q(imax+1); // val = 4qi
```

```
Selon (imax) :
```

```
    Cas imax=0 :
```

```
        q(2) = (R(3,2) - R(2,3))/val;
```

```
        q(3) = (R(1,3) - R(3,1))/val;
```

```
        q(4) = (R(2,1) - R(1,2))/val;
```

```
    Cas imax=1 :
```

```
        q(1) = (R(3,2) - R(2,3))/val;
```

```
        q(3) = (R(2,1) + R(1,2))/val;
```

```
        q(4) = (R(1,3) + R(3,1))/val;
```

```
    Cas imax=2 :
```

```
        q(1) = (R(1,3) - R(3,1))/val;
```

```
        q(2) = (R(2,1) + R(1,2))/val;
```

```
        q(4) = (R(3,2) + R(2,3))/val;
```

```
    Cas imax=3 :
```

```
        q(1) = (R(2,1) - R(1,2))/val;
```

```
        q(2) = (R(1,3) + R(3,1))/val;
```

```
        q(3) = (R(3,2) + R(2,3))/val;
```

```
Normalize(q);
```

Cet algorithme est assez économique en terme d'opération (une seule opération lourde de racine carrée sqrt).

2. Passage du quaternion à la matrice de rotation

On sait qu'une rotation d'axe unitaire \mathbf{v} et d'angle θ est représentée par le quaternion :

$$\mathbf{q} = \cos \frac{\theta}{2} + \mathbf{v} \sin \frac{\theta}{2} = \left(\cos \frac{\theta}{2}, x_v \sin \frac{\theta}{2}, y_v \sin \frac{\theta}{2}, z_v \sin \frac{\theta}{2} \right).$$

Au lieu de récupérer l'angle et les axes puis de construire la matrice de rotation (gourmand en terme d'opérations), on va utiliser l'expression de la matrice donnée au paragraphe précédent.

QuaternionToMatrix(Quaternion \mathbf{q} , Matrix \mathbf{R})

double u11, u12, u13, u14;

double u22, u23, u24 ;

double u33, u34 ;

double u44 ;

u11 = $\mathbf{q}(1) * \mathbf{q}(1)$; u12 = $\mathbf{q}(1) * \mathbf{q}(2)$;

u13 = $\mathbf{q}(1) * \mathbf{q}(3)$; u14 = $\mathbf{q}(1) * \mathbf{q}(4)$;

u22 = $\mathbf{q}(2) * \mathbf{q}(2)$; u23 = $\mathbf{q}(2) * \mathbf{q}(3)$; u24 = $\mathbf{q}(2) * \mathbf{q}(4)$;

u33 = $\mathbf{q}(3) * \mathbf{q}(3)$; u34 = $\mathbf{q}(3) * \mathbf{q}(4)$;

u44 = $\mathbf{q}(4) * \mathbf{q}(4)$;

$\mathbf{R}(1,1) = 2 * (u11 + u22) - 1$;

$\mathbf{R}(1,2) = 2 * (u23 + u14)$;

$\mathbf{R}(1,3) = 2 * (u24 - u13)$;

$\mathbf{R}(2,1) = 2 * (u23 - u14)$;

$\mathbf{R}(2,2) = 2 * (u11 + u33) - 1$;

$\mathbf{R}(2,3) = 2 * (u34 + u12)$;

$\mathbf{R}(3,1) = 2 * (u24 + u13)$;

$\mathbf{R}(3,2) = 2 * (u34 - u12)$;

$\mathbf{R}(3,3) = 2 * (u11 + u44) - 1$;

3. Multiplication de deux quaternions

En partant des propriétés spécifiques de la base des quaternions (voir chapitre 2), la multiplication de deux quaternions quelconques $\mathbf{q} = (q_1, q_2, q_3, q_4)$ et $\mathbf{p} = (p_1, p_2, p_3, p_4)$ est :

$$\begin{aligned} \mathbf{q} * \mathbf{p} = & p_1 q_1 - p_2 q_2 - p_3 q_3 - p_4 q_4 \\ & + (p_1 q_2 + p_2 q_1 + p_3 q_4 - p_4 q_3) \cdot \mathbf{i} \\ & + (p_1 q_3 + p_3 q_1 - p_2 q_4 + p_4 q_2) \cdot \mathbf{j} \\ & + (p_1 q_4 + p_4 q_1 + p_2 q_3 - p_3 q_2) \cdot \mathbf{k} \end{aligned}$$

Quant à la fonction utilisée à l'annexe 1, cela donne (après simplification car ici le premier quaternion est un vecteur de \mathbb{R}^3 sa partie réelle est donc nulle $p_1 = 0$) :

```
QuaternionMul(Vector u, Quat q)
```

```
Quat p;
```

```
p(1) = -u(1)*q(2) - u(2)*q(3) - u(3)*q(4);
```

```
p(2) = u(1)*q(1) + u(2)*q(4) - u(3)*q(3);
```

```
p(3) = u(2)*q(1) - u(1)*q(4) + u(3)*q(2);
```

```
p(4) = u(3)*q(1) + u(1)*q(3) - u(2)*q(2);
```

```
retourne p;
```

Annexe 3

Normales

Bien que la normale à la surface en un sommet \mathbf{p}_i d'une surface triangulée ne soit pas mathématiquement définie, il arrive fréquemment en infographie d'avoir besoin d'une valeur en chacun des sommets. Cela est par exemple nécessaire dans tous les formats d'affichage 3D lorsqu'on veut faire du rendu (avec une méthode de Gouraud ou de Phong), ou encore, on l'a vu dans cette thèse, lorsque l'on doit calculer le volume ou vérifier la contrainte d'incompressibilité. On choisit alors une expression de la moyenne des normales voisines le sommet \mathbf{p}_i .

Le but de cet annexe est de démontrer que quelle que soit l'expression choisie, lorsque l'élément de surface est le triangle, on désigne la même direction. On pourra se référer à des articles généraux sur les normales à une surface discrète, notamment (Glassner, 1990) et (Tampieri, 1992).

Trois choix sont possibles (voir Figure 1) :

1. \mathbf{n}_i : moyenne des normales aux facettes concourantes en \mathbf{p}_i , Figure 1.a :

$$\mathbf{n}_i = \frac{1}{nbf(i)} \sum_{j=1}^{nbf(i)} \mathbf{n}_{F(j)},$$

où $F(j)$ est la $j^{\text{ième}}$ facette concourante en \mathbf{p}_i , et $nbf(i)$ le nombre de facettes concourantes en \mathbf{p}_i et $\mathbf{n}_{F(j)}$ la normale à $F(j)$. $nbf(i)$ est aussi le nombre de voisins de \mathbf{p}_i car les facettes sont triangulaires (et on suppose la surface fermée).

2. \mathbf{n}'_i : moyenne des produits vectoriels deux à deux de $\mathbf{p}_i \mathbf{p}_{v(i,j)} \times \mathbf{p}_i \mathbf{p}_{v(i,j \oplus 1)}$, Figure 1.b :

$$\mathbf{n}'_i = \frac{1}{nbf(i)} \sum_{j=1}^{nbf(i)} \mathbf{p}_i \mathbf{p}_{v(i,j)} \times \mathbf{p}_i \mathbf{p}_{v(i,j \oplus 1)},$$

où $\mathbf{p}_{v(i,j)}$ est le $j^{\text{ième}}$ voisin de \mathbf{p}_i (\oplus dénote l'addition modulo $nbf(i)$).

3. \mathbf{n}_i'' : moyenne des produits vectoriels deux à deux de $\mathbf{p}_{v(i,j)} \times \mathbf{p}_{v(i,j\oplus 1)}$, Figure 1.c :

$$\mathbf{n}_i'' = \frac{1}{\text{nbf}(i)} \sum_{j=1}^{\text{nbf}(i)} \mathbf{p}_{v(i,j)} \times \mathbf{p}_{v(i,j\oplus 1)}.$$

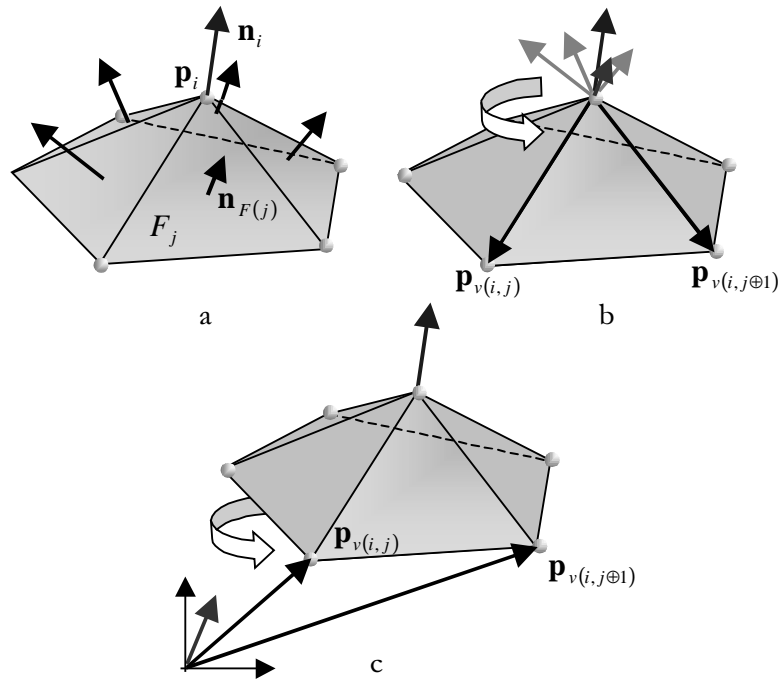


Figure 1. Trois façons d'exprimer la normale en un sommet d'une surface triangulée.

1.1.1 \mathbf{n}_i colinéaire à \mathbf{n}_i'

$\mathbf{n}_i = \sum_{j=1}^{\text{nbf}(i)} \mathbf{n}_{F(j)}$, mais comme les facettes sont triangulaires, on a :

$$\mathbf{n}_{F(j)} = \frac{\mathbf{p}_i \mathbf{p}_{v(i,j)} \times \mathbf{p}_i \mathbf{p}_{v(i,j\oplus 1)}}{\|\mathbf{p}_i \mathbf{p}_{v(i,j)} \times \mathbf{p}_i \mathbf{p}_{v(i,j\oplus 1)}\|}$$

On a donc :

$$\begin{aligned} \mathbf{n}_i &= \frac{1}{nbf(i)} \sum_j \mathbf{n}_{F(j)} = \frac{1}{nbf(i)} \sum_j \frac{\mathbf{p}_i \mathbf{p}_{v(i,j)} \times \mathbf{p}_i \mathbf{p}_{v(i,j\oplus 1)}}{\|\mathbf{p}_i \mathbf{p}_{v(i,j)} \times \mathbf{p}_i \mathbf{p}_{v(i,j\oplus 1)}\|} \\ &= \frac{1}{nbf(i)} \frac{\sum_j \mathbf{p}_i \mathbf{p}_{v(i,j)} \times \mathbf{p}_i \mathbf{p}_{v(i,j\oplus 1)}}{\sum_j \|\mathbf{p}_i \mathbf{p}_{v(i,j)} \times \mathbf{p}_i \mathbf{p}_{v(i,j\oplus 1)}\|} = \frac{1}{nbf(i) \cdot \sum_j \|\mathbf{p}_i \mathbf{p}_{v(i,j)} \times \mathbf{p}_i \mathbf{p}_{v(i,j\oplus 1)}\|} \mathbf{n}'_i \\ &= k_1 \cdot \mathbf{n}'_i \end{aligned}$$

C'est-à-dire que \mathbf{n}_i est colinéaire à \mathbf{n}'_i .

1.1.2 \mathbf{n}'_i colinéaire à \mathbf{n}''_i (voir aussi chapitre 4)

$$\begin{aligned} \mathbf{n}'_i &= \frac{1}{nbf(i)} \sum_{j=1}^{nbf(i)} \mathbf{p}_i \mathbf{p}_{v(i,j)} \times \mathbf{p}_i \mathbf{p}_{v(i,j\oplus 1)} \\ &= \frac{1}{nbf(i)} \sum_{j=1}^{nbf(i)} (\mathbf{p}_{v(i,j)} \times \mathbf{p}_{v(i,j\oplus 1)} + \mathbf{p}_i \times \mathbf{p}_{v(i,j)} \mathbf{p}_{v(i,j\oplus 1)}) \end{aligned}$$

Quand on fait la sommation sur l'ensemble des facettes concourantes :

$$\sum_{j=1}^{nbf(i)} \mathbf{p}_i \times \mathbf{p}_{v(i,j)} \mathbf{p}_{v(i,j\oplus 1)} = \mathbf{p}_i \times \sum_{j=1}^{nbf(i)} \mathbf{p}_{v(i,j)} \mathbf{p}_{v(i,j\oplus 1)} = \mathbf{p}_i \times \mathbf{0} = \mathbf{0},$$

où $\mathbf{0}$ dénote le vecteur nul. On en déduit que :

$$\mathbf{n}'_i = \frac{1}{nbf(i)} \sum_j \mathbf{p}_{v(i,j)} \times \mathbf{p}_{v(i,j\oplus 1)} = \mathbf{n}''_i.$$

Annexe 4

Format du modèle

Le format des objets mis au point pour être utilisé dans notre modèle vérifie les principes suivant :

- Il doit être indépendant de l'affichage,
- Il doit pouvoir être édité à la main sans risque,
- Il doit être suffisamment ouvert pour accepter des nouvelles données en cours de développement.

Nous avons donc défini une grammaire du format. Nous donnons ici le fichier descriptif de ce format et un exemple d'objet.

1. Description du format :

"#" denotes comments (up to the end of line)

Global or regional force weights will be inherited by each element unless a new global, regional or element weight is specified.

Each element of a region is of the same type, unique id for region and elem are needed (id can correspond to none element or region). Element and region id could begin at index 0.

In the following grammar, **i** denotes an int value, and **f** a float value, each instruction should be ended by an end of line character :

<object>	:=	nbElem i <object>	<i>the object nr of elements</i>
		nbRegion i <object>	<i>the object nr of regions</i>
		externW f <object>	<i>the extern force weight parameter</i>
		viscosityW f <object>	<i>the viscosity force weight parameter</i>
		beginRegion i <region><object>.	<i>begin region i (i is an unique region id)</i>

<region>	:=	type elastic <elasticRegion>	<i>this is an elastic region</i>
		type muscular <muscularRegion>	<i>this is a muscular region</i>
		type solid <solidRegion>.	<i>this is a solid region</i>

<genRegion> := nbElem *i* <genRegion> | *gives the nr of element*
 externW *f* <genRegion> | *gives the extern weight paramater*
 viscosityW *f* <genRegion> | *gives the viscosity weight parameter*
 mass *f* <genRegion> | *each element of this region has this mass*
 endRegion. *end of the current region*

<elasticRegion> := shapeW *f* <elasticRegion> | *shape memory force weight parameter*
 masterRegion {yes|no} <elasticRegion> | *sets this region as the master region*
 setAssociate *i* <elasticRegion> | *region i is an associate of current region*
 cteVol {yes|[no]} <elasticRegion> | *sets or not the constant volume constraint*
 beginElem *i* <elasticElem> | *begins element i (unique element id)*
 <genRegion>.

<muscularRegion> := muscularW *f* <muscularRegion> | *muscular force weight*
 autoMuscleNeigh {yes|[no]} <muscularRegion> | *automatically sets the neighbours as muscle contraction direction*
 ONLY for the neighbours that are in the same muscular region
 beginElem *i* <muscularElem> | *begins element i (unique element id)*
 <elasticRegion>.

<solidRegion> := linkW *i* <genRegion> | *weight of springs between solid regions*
 beginElem *i* <solidElem> | *begins element i (unique element id)*
 <genRegion>.

<genElem> := coord *f f f* <genElem> | *x, y and z position of the element*
 externW *f* <genElem> | *external force weight parameter*
 viscosityW *f* <genElem> | *viscosity force weight parameter*
 mass *f* <genElem> | *element mass*
 neigh *i* <genElem> | *elem i is the next neighbour*
 (becareful :the neighbour order is to be the clockwise order)
 link *i* <genElem> | *if in elastic or muscular element, i is the solid element that*
constrains the current element. If in solid element i is the solid element
linked by a spring to the current element.
 fixed {yes|[no]} <genElem> | *fixed the element (by default not fixed)*
 endElem.

$\langle \text{elasticElem} \rangle := \text{shapeW } f \langle \text{elasticElem} \rangle \mid \text{shapeMemory } \{[\text{yes}] \mid \text{no}\} \mid$
shape memory force weight parameter
the elastic force is computed either by shape memory
or mass-spring method
 $\langle \text{genElem} \rangle.$

$\langle \text{muscularElem} \rangle := \text{muscularW } f \langle \text{muscularElem} \rangle \mid \text{notMuscleNeigh } i \langle \text{muscularElem} \rangle \mid$
muscular force weight
neighbour i does NOT define a contraction direction
 $\langle \text{elasticElem} \rangle.$

$\langle \text{solidElem} \rangle := \text{linkW } f \langle \text{genElem} \rangle \mid$
force weight of spring for the solid-solid link
this corresponds to the spring stiffness of the link
 $\langle \text{genElem} \rangle.$

2. Exemple

L'exemple choisi est un cube dont une face est définie par une région solide, la face opposée par une région musculaire et la couronne par une région élastique (voir Figure 1).

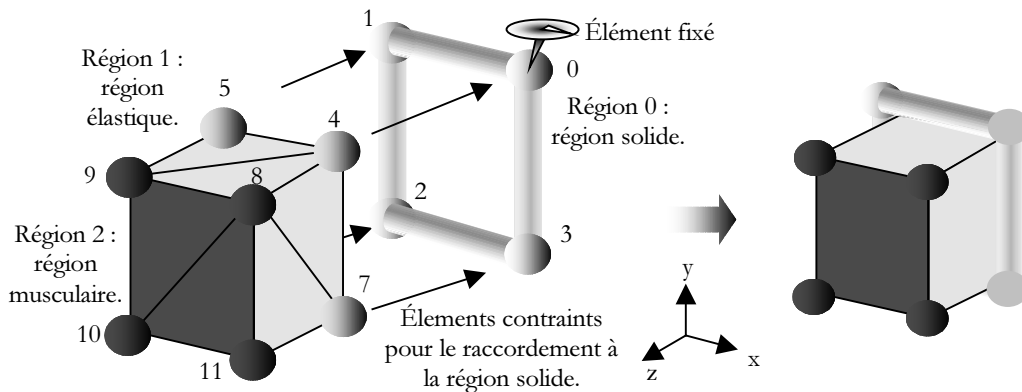


Figure 1. Schéma d'un objet utilisable par le modèle. La région élastique occupe le pourtour du cube, la région musculaire la face antérieure.

Les régions élastiques et musculaires définissent le volume du cube qui doit rester constant. Pour cela on utilise la contrainte de volume constant. C'est la région élastique qui pilote la contrainte (masterRegion).

On utilise la fonction `autoMuscleNeigh` afin de définir les directions de contractions seulement dans la direction des éléments voisins musculaires (par défaut, tous les voisins sont musculaires). On aurait pu aussi enlever chacun des voisins élastiques de la liste des voisins musculaires, un par un.

Voici le fichier résultant :

```
# Fichier de description
nbRegion 3
nbElem 12
externW 9.81
viscosityW 1.
```

```
# région solide
```

```
beginRegion 0
  type solid
  nbElem 4
  mass 10.0

  beginElem 0
    coord 1 1 0
    neigh 1
    neigh 3
    fixed yes
  endElem

  beginElem 1
    coord 0 1 0
    neigh 0
    neigh 2
  endElem

  beginElem 2
    coord 0 0 0
    neigh 1
    neigh 3
  endElem

  beginElem 3
    coord 1 0 0
    neigh 2
    neigh 0
  endElem
endRegion
```

```
# région élastique
```

```
beginRegion 1
  type elastic
  nbElem 4
  mass 5.0
  shapeW 10.0
  cteVol yes
  masterRegion yes
  setAssociate 2

  beginElem 4
    coord 1 1 0
    link 0
    neigh 5
    neigh 9
    neigh 8
    neigh 7
  endElem

  beginElem 5
    neigh 6
    neigh 10
    neigh 9
    neigh 4
    coord 0 1 0
    link 1
  endElem

  beginElem 6
    coord 0 0 0
    neigh 7
    neigh 11
    neigh 10
    neigh 5
    link 2
  endElem

  beginElem 7
    coord 1 0 0
    link 3
    neigh 4
    neigh 8
    neigh 11
    neigh 6
  endElem
endRegion
```

```
# région musculaire
```

```
beginRegion 2
  type muscular
  nbElem 4
  mass 1.0
  shapeW 3.0
  cteVol yes
  masterRegion no
  autoMuscleNeigh yes

  beginElem 8
    coord 1 1 1
    neigh 4
    neigh 9
    neigh 10
    neigh 11
    neigh 7
  endElem

  beginElem 9
    coord 0 1 1
    neigh 5
    neigh 10
    neigh 8
    neigh 4
  endElem

  beginElem 10
    coord 0 0 1
    neigh 5
    neigh 6
    neigh 11
    neigh 8
    neigh 9
  endElem

  beginElem 11
    coord 1 0 1
    neigh 7
    neigh 8
    neigh 10
    neigh 6
  endElem
endRegion
```

Bibliographie

- (Adler et al, 1996) A. Adler, R. Guardo, Y. Berthiaume. Impedance imaging of lung ventilation : do we need to account for chest expansion ? *IEEE Transactions on biomedical engineering*. 43(4), pp 414-420. Avril 1996.
- (Agostini et Hyatt, 1986) E. Agostoni, R.E. Hyatt. Static behavior of the respiratory system. Dans (Macklem et Mead, 1986), chapitre 9, pp 15-34.
- (Aubert et Bechmann, 1995) F. Aubert, D. Bechmann. Déformation d'objets à volume constant. Actes de l'AFIG, pp 149-158. Novembre 1995, identique à rapport 95/09, Laboratoire des Sciences de l'Image et de l'Informatique et de la Télédétection, Université Louis Pasteur, Strasbourg.
- (Aubert et Bechmann, 1997) F. Aubert, D. Bechmann. Volume preserving space deformation. *Computer & Graphics*. 21(5), pp . Septembre-octobre 1997.
- (Ayache et al., 1996) N. Ayache, P. Cinquin, I. Cohen, L. Cohen, F. Leitner, O.Monga. Segmentation of complex medical objects : a challenge and a requirement for computer assisted surgery planning and performing. In R. Taylor, S. Lavallée, G. Burdea and R. Mosges, editors. *Computer Integrated Surgery*. MIT Press, Cambridge, MA. 1996.
- (Badler, 1997) N. Badler. Virtual humans for animation, ergonomics and simulation. *IEEE Workshop on non-rigid and articulated motion*. Puerto Rico, Juin 1997.
- (Bainville, 1992) E. Bainville. *Reconstruction d'images tridimensionnelles à partir de silhouettes*. DEA informatique. Université Joseph Fourier. 1992.

- (Bainville, 1996) E. Bainville. *Modélisation géométrique et dynamique d'un geste chirurgical*. Thèse de l'Université Joseph Fourier, Grenoble. Mars 1996.
- (Baraff, 1994) D. Baraff. Fast contact force computation for nonpenetrating rigid bodies. *Computer Graphics*, 28, pp 23-34. Juillet 1994. Siggraph 94.
- (Barraf et Witkin, 1992) D. Baraff, A. Witkin. Dynamic simulation of nonpenetrating flexible bodies. *Computer Graphics*, 26(2), pp 303-308. Juillet 1992. Siggraph 92.
- (Barzel et Barr, 1988) R. Barzel, A.H. Barr. A modeling system based on dynamics constraints. *Computer Graphics*, 22(4), pp 179-188. Août 1988. Siggraph 88.
- (Bausset, 1982) M. Bausset. *Dynamiques : principes classiques et relativistes, milieux solides et déformation*. Hermann. 1982.
- (Bechmann et Dubreuil, 1993) D. Bechmann et N. Dubreuil. Animation through space and time based on a space deformation model. *The Journal of Visualisation and Computer Animation*. 4(3), pp 165-184. Juillet-septembre 1993.
- (Ben-Haim et Saidel, 1990) S.A. Ben-Haim, G. Saidel. Mathematical model of chest wall mechanics : a phenomenological approach. *Annal of Biomedical Engineering*. 18, pp 37-56. 1990
- (Beylot et al, 1996) P. Beylot, P. Gingsins, P. Kalra, N. Magnenat Thalmann, W. Maurel, D. Thalmann, J. Fasel. 3D interactive topological modeling using visible human dataset. *Eurographics '96*. Août 1996.
- (Bittar et al, 1993) E. Bittar, S. Lavallee, R. Szeliski. Method for registering overlapping range images of arbitrarily shaped surfaces for 3D object reconstruction. SPIE. Sensor Fusion IV, Boston. Vol 2059, pp 384-395. 1993.
- (Borrel et Bechmann, 1991) P. Borrel, D. Bechmann. Deformation of n-dimensional objects. *International Journal of Computational Geometry & Applications*. 1(4), pp 427-453. 1991.
- (Boulic et al, 1990) R. Boulic, N. Magnenat-Thalmann, D. Thalmann. A global human walking model with real-time kinematic personification. *The Visual Computer*. 6(6), pp. 344-358. 1990.
- (Bruderlin et Calvert, 1989) A. Bruderlin, T.W. Calvert. Goal-directed, dynamic animation of human walking. *Computer Graphics 1989 Proceedings*. 23, pp 233-242. Juillet 1989. Siggraph'89.

- (Bruderlin et Calvert, 1993) A. Bruderlin, T.W. Calvert. Interactive animation of personalized human locomotion. *Graphics Interface 1993 proceedings*. Pp 17-23
- (Burdea et Coiffet, 1993) G. Burdea, P. Coiffet. *La réalité virtuelle*. Hermès, Paris 1993.
- (Butler et Leith, 1986) J.P. Butler, D.D. Leith. Principles of measurement : applications to pressure, volume, and flow. Dans (Macklem et Mead, 1986), chapitre 2, pp 15-34.
- (Carnevali et al, 1996) P. Carnevali, G. Ferrigno, A. Aliverti, A. Pedotti. A new method for 3D optical analysis of chest wall motion. *Technology and health care*. 4, pp 43-65. 1996.
- (Chadwick et al., 1989) J.E. Chadwick, D.R. Hauman, R.E. Parent. Layered construction for deformable animated characters. *Computer Graphics*, 23(4), pp 243-252. Juillet 1989. Siggraph 89.
- (Chaudy, 1997) C. Chaudy. *Synthèse d'images réalistes de paysages naturels*. Thèse de l'université Joseph Fourier, Grenoble. Février 1997.
- (Chen, 1992) D.T.-W. Chen. *Pump it up : computer animation of a biomechanically based model of muscle using the finite element method*. Thèse du Massachusetts Institute of Technology. Février 1992.
- (Chen et Zeltzer, 1992) D.T. Chen, D. Zeltzer. Pump it up : computer animation of a biomechanically based model of muscle using the finite element method. *Computer Graphics*, 26(2), pp 89-98. Juillet 1992. Siggraph'92.
- (Ciarlet, 1985) P.G. Ciarlet. *Elasticité tridimensionnelle*. Masson, 1985.
- (Ciarlet, 1988) P.G. Ciarlet. *Introduction to numerical linear algebra and optimization*. Cambridge, England : Cambridge University Press. 1988.
- (Cinquin et al, 1995) P. Cinquin et al. Computer assisted medical interventions at TIMC laboratory : passive and semi-active aids. *IEEE engineering in Medicine and Biology magazine, special issue Robots in Surgery*. 14(3) : pp 254-263. 1995.
- (Closkey et al, 1992) R. F. Closkey, A.B. Schultz, C.W. Luchies. A model for studies of the deformable rib cage. *Journal of Biomechanics*. 25(5), pp 529-539. 1992.
- (Cotin et al, 1996) S. Cotin, H. Delingette, M. Bro-Nielsen, N. Ayache, J.M. Clément, V. Tassetti, J. Marescaux. Geometric and physical representations for a simulator of hepatic surgery. *Health Care in the Information Age*, H. Sieburg, S. Weghorst, et K. Morgan (Eds), IOS Press and Ohmsha. 1996. *Medicine Meets Virtual Reality 4*.

- (Cozot et al., 1995) R. Cozot, B. Arnaldi, G. Dumont. A unified model for physically based animation and simulation. In *Applied Modelization, Simulation and Optimisation.*, Cancun, Mexique. Juin 1995.
- (Delingette, 1994) H. Delingette. *Simplex meshes : a general representation for 3D shape reconstruction*. Technical report 2214. Inria. Mars 1994.
- (Delnondedieu et al, 1993) Y. Delnondedieu, A. Luciani, C. Cadoz. Physical elementary component for modeling the tensory-motricity : the primary muscle. *4th Eurographics Computer Animation and Simulation Workshop*. Septembre 1993.
- (Desbrun et Gascuel, 1994) M. Desbrun, M.P. Gascuel, Highly deformable material for animation and collision processing. *5th Eurographics Workshop on Animation and Simulation*, pp 89-102, Oslo, Norvège. Septembre 1994.
- (De Troyer et Loring, 1986) A. De Troyer, S.H. Loring. Action of the respiratory muscles. Dans (Macklem et Mead, 1986), chapitre 26, pp 443-461.
- (Dumont et al, 1989) G. Dumont, B. Arnaldi, G. Hégron. Mechanics of solids for computer animation. *Proceedings PIXIM 89*, 1989.
- (Edelstein-Keshet, 1988) L. Edelstein-Keshet, *Mathematical models in biology*. Birkhauser mathematics series. Mc Graw Hill. 1988.
- (Faure, 1996) F. Faure. An energy-based approach for contact force computation. In *Computer Graphics Forum*, 15(3). Août 1996. Eurographics 96.
- (Ferrigno et al, 1990) G. Ferrigno, N.A. Borhese, A. Pedotti. Pattern recognition in 3D automatic human analysis. *ISPRS Journal of photogrammetry and remote sensing*. 45, pp 224-246. 1990.
- (Ferrigno et al, 1994) G. Ferrigno, P. Carnevali, A. Aliverti, F. Molteni, G. Beulcke, A. Pedotti. Three-dimensional optical analysis of chest wall motion. *Journal of Applied Physiology*. 77(3), pp 1224-1231. 1994.
- (Foley et al., 1990) J.D. Foley, A. van Dam, S.K. Feiner, J.F. Hughes. *Computer Graphics, principle and practice*. 2nd edition. Addison-Wesley, 1990.
- (Fua et Bechbühler, 1997) P. Fua, C. Brechbühler. Imposing hard constraints on soft snakes. *Computer vision and image understanding*. 65(2), pp 148-162. 1997. Voir aussi : *Technical note 553*, AI center SRI international. Octobre 1995.

- (Garat, 1996) F. Garat. *Identification de paramètres physiques pour un simulateur dynamique appliqué à la robotique*. DEA informatique. Institut National Polytechnique de Grenoble. Juin 1996.
- (Gascuel, 1990) M.P. Gascuel. *Déformations de surfaces complexes : techniques de haut niveau pour la modélisation et l'animation*. Thèse de l'université de Paris-Sud, Centre d'Orsay. octobre 1990.
- (Gascuel, 1993) M.P. Gascuel. An implicit formulation for precise contact modeling between flexible solids. *Computer Graphics*, pp 313-320, Août 93. Siggraph 1993.
- (Gascuel et Desbrun, 1997) M.P. Gascuel, M. Desbrun. Animation of deformable models using implicit surfaces. *IEEE Transactions on Visualization and Computer Graphics*, 3(1). Mars 1997.
- (Gascuel et Gascuel, 1994) J.D. Gascuel, M.P. Gascuel. Displacement constraints for interactive modeling and animation of articulated structures, *The Visual Computer*, 10(4), pp 191-204. Mars 1994.
- (Gascuel et Puech, 1994) M.P. Gascuel, C. Puech. Dynamic animation of deformable bodies. *From object modelling to advanced visual communication*, éd. S. Coquillart, W. Straber, P. Stucki, série *Focus on Computer Graphics*, pp 118-139, Springer-Verlag 1994.
- (Gascuel, 1995) M.P. Gascuel. *Modèles déformables pour l'animation : modélisation, animation et contrôle*. Mémoire d'habilitation à diriger des recherches de l'Institut National Polytechnique de Grenoble, mars 1995.
- (Gaskins, 1992) T. Gaskins. *PEXlib programming manual*. O'Reilly & Associates, 1992.
- (Gini et Gini, 1985) G. Gini, M. Gini. *Robotique : contrôle, programmation, interaction avec l'environnement*. Masson. 1985.
- (Glassner, 1990) A. S. Glassner. Computing surface normals for 3D models. *Graphics Gems I*, A. S. Glassner ed. Academic Press, 1990.
- (Goldman, 1991) R. N. Goldman. Area of planar polygons and volume of polyhedra. *Graphics Gems II*, J. Arvo ed. Academic Press, 1991.
- (Gourret et al., 1989) J.P. Gourret, N.M. Thalmann, D. Thalmann. Simulation of object and human skin deformations in a grasping task. *Computer Graphics*, 23(3), pp 21-30. Juillet 1989. Siggraph 89.
- (Grassino et Goldman, 1986) A.E. Grassino, M.D. Goldman. Respiratory muscle coordination, Dans (Macklem et Mead, 1986), chapitre 27, pp 463-480.

- (Grzeszczuk et Terzopoulos, 1995) R. Grzeszczuk, D. Terzopoulos. Automated learning of muscle-actuated locomotion through control abstraction. *Computer Graphics 1995 proceedings*, pp 63-70. Août 1995. Siggraph 95.
- (Hodgins et al, 1995) J.K. Hodgins, W.L. Wooten, D.C. Brogan, J.F. O'Brien. Animating human athletics. *Computer Graphics, 1995 proceedings*, pp 71-78. Août 1995. Siggraph 95.
- (Isaacs et Cohen, 1987) P.M. Isaacs, M.F. Cohen. Controlling dynamic simulation with kinematic constraints, behavior functions and inverse dynamics. *Computer Graphics*, 21(4), pp 215-224. Juillet 1987. Siggraph 87.
- (Jimenez, 1993) S. Jimenez. *Modélisation et simulation physique d'objets volumiques déformables complexes*. Thèse de l'Institut National Polytechnique de Grenoble. Novembre 1993.
- (Joukhadar, 1997) A. Joukhadar. *Simulation dynamique et applications robotiques*. Thèse de l'Institut Polytechnique National de Grenoble. mai 1997.
- (Kass et al, 1988) M. Kass, A. Witkin, D. Terzopoulos. Snakes : active contour models. *International Journal of Computer Vision*, pp 321-331. 1988.
- (Kaye et al., 1995) J.K. Kaye, F.P. Primiano, D. Metaxas. Anatomical and physiological simulation for respiratory mechanics. *Journal of Image Guided Surgery*. 1, pp 164-171. 1995. Identique à proceedings of Medicine Meets Virtual Reality 4, 1996 et proceedings of Medical Robotics and Computer Assisted Surgery 1995.
- (Kaye et al., 1997) J. Kaye, D. N. Metaxas, F.P. Primiano Jr. A 3D virtual environment for modeling mechanical cardiopulmonary interactions. CVRMed II-MRCAS III 1st joint conference. (Lecture notes in CS, vol. 1205, pp 389-398. Springer Verlag). Mars 1997. Mars 1997.
- (Kilgard, 1996) M. J. Kilgard. *OpenGL Programming for the X window System*. Addison-Wesley Publishing Company, 1996.
- (Konno et Mead, 1967) Measurement of separate volume changes of rib cage and abdomen during breathing. *Journal of Applied Physiology*. 22. 407-422. 1967
- (Lachaud et Bainville, 1994) J.O. Lachaud, E. Bainville. A discrete adaptative model following topological modifications of volumes. *Proceedings of Discrete geometry for computer imagery*. Grenoble. Septembre 1994.

- (Lamouret, 1995) A. Lamouret. *Animation par modèles générateurs : contrôle du mouvement*. Thèse de l'université Joseph Fourier, Grenoble. Septembre 1995.
- (Lamouret et al, 1995) A. Lamouret, M.P. Gascuel, J.D. Gascuel. Combining physically-based simulation of colliding objects with trajectory control. *The Journal of Visualization and Computer Animation*, 5, pp 1-20. 1995.
- (Larralde, 1986) J.P. Larralde. *Dynamique*. Masson. 1986.
- (Lascaux et Théodor, 1994) P. Lascaux, R. Théodor. *Analyse numérique matricielle appliquée à l'art de l'ingénieur*. 2nde édition. Masson. 1994.
- (Lee et al., 1995) Y. Lee, D. Terzopoulos, K. Waters. Realistic modeling for facial animation. *Computer Graphics 1995 proceedings*, pp 55-62. Août 1995. Siggraph 95.
- (Leitner, 1993) F. Leitner. *Segmentation dynamique d'images tridimensionnelles*. Thèse de l'Institut National Polytechnique de Grenoble. Septembre 1993.
- (Lombardo et Puech, 1995) J.C. Lombardo, C. Puech. Oriented particles : a tool for shape memory objects modelling. *Graphics Interface '95*, pp 255-262. Quebec city, Canada. 1995.
- (Lombardo, 1996) J.C. Lombardo. *Modélisation d'objets déformables avec un système de particules orientées*. Thèse de l'université Joseph Fourier, Grenoble. Janvier 1996.
- (Loring et Bruce, 1986) S.H. Loring, E.N. Bruce. Methods for study of the chest wall., Dans (Macklem et Mead, 1986), chapitre 24, pp 415-428.
- (Luciani et Cadoz, 1986) A. Luciani, C. Cadoz. Utilisation de modèles mécaniques et géométriques pour la synthèse et le contrôle d'images animées. *Deuxième colloque Image. CESTA*. Nice, avril 1986.
- (Macklem et Mead, 1986) P.T. Maclem, J. Mead. The respiratory system, *Handbook of Physiology*, vol III, Part I et II, American Physiological society, 1986.
- (Magnenat-Thalman et Thalman, 1987) N. Magnenat-Thalman, D. Thalman. The direction of synthetic actors in the film *Rendez-vous à Montréal*. *IEEE Computer Graphics and applications*. 7(12), pp 9-19. Décembre 1987.
- (Magnenat-Thalman et Thalman, 1991) N. Magnenat-Thalman, D. Thalman. Complex models for animating synthetic actors. *IEEE Computer Graphics and Applications*. 11(5), pp 32-44.

- (Mantyla, 1988) M. Mantyla, *An introduction to solid modeling*. Computer Science Press. 1988.
- (Maurel et al, 1996) W. Maurel, D. Thalmann, P. Hoffmeyer, P. Beylot, P. Gingins, P. Kalra, N. Magnenat Thalmann. A biomechanical musculoskeletal model of human upper limb for dynamic simulation. *Eurographics Computer Animation and Simulation Workshop 1996*, pp 121-136. Août 1996.
- (Mead et Loring, 1982) J. Mead, S.H. Loring. Analysis of volume displacement and length changes of the diaphragm during breathing. *Journal of Applied Physiology Respiration Environment. Exercice Physiology*. 53. Pp 750-755. 1982.
- (Meseure, 1997) P. Meseure. *Modélisation de corps déformables pour la simulation d'actes chirurgicaux*. Thèse de l'université des sciences et technologies de Lille. Janvier 1997.
- (Miller, 1988) G.S.P. Miller. The motion dynamics of snakes and worms. *Computer Graphics*, 22(4). Août 1988. Siggraph 88.
- (Moore et Wilhelms, 1988) M. Moore, J. Wilhelms. Collision detection and response for computer animation. *Computer Graphics*, 22(4), pp 289-298. Août 1988. Siggraph 88.
- (Neider et al., 1993) J. Neider, T. Davis, M. Woo. *OpenGL Programming Guide*. Addison-Wesley Publishing Company, 1993. ISBN 0-201-63274-8. <http://www.sgi.com/Technology/OpenGL>
- (Otis, 1986) A.B. Otis. History of respiratory mechanics. Dans (Macklem et Mead, 1986), chapitre 1, pp 1-12.
- (van Overveld, 1991) C.W.A.M. van Overveld. An iterative approach to dynamic simulation of 3-D rigid-body motions for real-time interactive computer animation. *The Visual Computer*, 7(1), pp 29-38. 1991.
- (Pentland et Williams, 1989) A. Pentland, J. Williams. Good vibrations : modal dynamics for graphics and animation. *Computer Graphics*, 23(3), pp 215-222. Juillet 1989. Siggraph 89.
- (Perrotin et Moreaux, 1965) J. Perrotin et J. Moreaux. *Chirurgie du diaphragme*. Monographie chirurgicales. Masson et Cie. Paris. 1965.
- (Petrol et al., 1990) W.M Petroll, H. Knight, D.F. Rochester. A model approach to assess diaphragmatic volume displacement, *Journal of Applied Physiology*, 69 (6), pp 2175-2182, 1990
- (Platt et Barr, 1988) J.C Platt, A.H. Barr. Constraint methods for flexible models. *Computer Graphics*, 22(4), pp 279-288. Août 1988. Siggraph 88.

- (Press et al, 1988) W.H. Press, B.P. Flannery, S.Q. Teukolsky, W.T. Vetterling. *Numerical recipes in C*. Cambridge University Press, 1988.
- (Primiano, 1982) F.P. Primiano Jr. Theoretical analysis of chest wall mechanics. *Journal of Biomechanics*, 15(12), pp 919-931. 1982.
- (Promayon et al, 1996) E. Promayon, P. Baconnier, C. Puech. Physically-based deformations constrained in displacements and volume. In *Computer Graphics Forum*, 15(3), pp 155-164. Août 1996. Eurographics 96.
- (Promayon et al, 1997) E. Promayon, P. Baconnier, C. Puech. Physically-based model for simulating the human trunk respiration movements. CVRMed II-MRCAS III 1st joint conference. (Lecture notes in CS, vol. 1205, pp 379-388. Springer Verlag). Mars 1997.
- (Provot, 1995) X. Provot. Deformation constraints in a mass-spring model to describe rigid cloth behavior. *Proceedings Graphics Interface '95*, pp 147-154. 1995.
- (Rappoport et al, 1996) A. Rappoport, A. Sheffer, M. Bercovier. Volume-preserving free-form solids. *IEEE Transactions on visualization and computer graphics*, 2(1), pp19-27. Mars 1996. Identique à Proceedings 3rd ACM/Siggraph symposium on solid modeling 95, May 1995, Salt Lake City, pp 351-372 et à A. Sheffer, Master of Science, Institute of Computer Science, Hebrew University of Jerusalem,.
- (Reeves, 1983) W.T. Reeves. Particle systems : a technique for modeling a classe of fuzzy objects. *ACM Transactions on Graphics*, 2(2), pp 91-108. Avril 92.
- (Rodarte et Rehder, 1986) J.R. Rodarte, K. Rehder. Dynamics of respiration. Dans (Macklem et Mead, 1986), chapitre 10, pp 131-144.
- (Roussos et Macklem, 1985) C. Roussos, P.T. Macklem. *The thorax (Part A & B)*, Lung biology in health and disease, éd. C. Lenfant, M. Dekker, 1985.
- (Sederberg et Parry, 1986) T.W. Sederberg, S.R. Parry. Free-form deformation of solid geometric models. *Computer Graphics*, 20(4), pp 151-160. Août 1986. Siggraph 86.
- (Sharp et Hyatt, 1986) J.T. Sharp, R.E. Hyatt. Mechanical and electrical properties of respiratory muscles. Dans (Macklem et Mead, 1986), chapitre 23, pp 389-414.

- (Shoemake, 1985) K. Shoemake. Animating rotation with quaternion curves. *Computer Graphics*, 19(3), pp 245-254. Juillet 1985. Siggraph 85.
- (Simon, 1996) D. Simon. *Souris 6D*. Rapport de stage de l'IUT informatique de Grenoble. 1996
- (Smith et Loring, 1986) J.C. Smith, S.H. Loring. Passive mechanical properties of the chest wall. Dans (Macklem et Mead, 1986), chapitre 25, pp 429-442.
- (Szeliski et Tonnesen, 1991) R. Szeliski, D. Tonnesen. *Surface modeling with oriented particle systems*. Digital Equipment Corporation, Cambridge research lab., CRL 91/14. Décembre 1991.
- (Szeliski et Tonnesen, 1992) R. Szeliski, D. Tonnesen. Surface modeling with oriented particle systems. *Computer Graphics*, 26(2), pp 185-194. Juillet 92. Siggraph 92.
- (Szilas, 1996) N. Szilas. *Apprentissage dans les réseaux récurrents pour la modélisation mécanique et étude de leurs interactions avec l'environnement*. Thèse de l'Institut National Polytechnique de Grenoble. Décembre 1996.
- (Tampieri, 1992) F. Tampieri. Newell's method for computing the plane equation of a polygon. *Graphic Gems III*, D. Kirk ed. Academic Press, 1992.
- (Taylor et al., 1989) A.E. Taylor, K. Rehder, R.R. Hyatt, J.C. Parker. *Clinical respiratory physiology*. Saunders, 1989.
- (Terzopoulos et al., 1987) D. Terzopoulos, J. Platt, A. Barr, K. Fleischer. Elastically deformable model. *Computer Graphics*, 21(4), pp 205-214. Juillet 87. Siggraph 87.
- (Terzopoulos et Fleischer, 1988) D. Terzopoulos, K. Fleischer. Modeling inelastic deformations : viscoelasticity, plasticity, fracture. *Computer Graphics*, 26(2), pp 269-278. Août 1988. Siggraph 88.
- (Terzopoulos et Metaxas, 1991) D. Terzopoulos, D. Metaxas. Dynamic 3D models with local and global deformations : deformable superquadrics. *IEEE Transactions on Pattern Analysis and machine Intelligence*, 13(7), pp 703-714. Juillet 1991.
- (Terzopoulos et al, 1989) D. Terzopoulos, J. Platt, A. Barr, D. Zeltzer, A. Witkin, J. Blinn. Physically-based modeling : past, present and future. *Siggraph 1989*. Panel session. Juillet 1989.
- (Terzopoulos et Witkin, 1988) D. Terzopoulos, A. Witkin. Physically-based models with rigid and deformable components, *IEEE Computer Graphics and Applications*, 8(6), 41-51. Novembre 1988.

- (Thalmann, 1996) D. Thalmann. The complexity of simulating virtual humans. EPFL, Super Computer Review. 8. Novembre 1996.
- (Volino et al, 1996) P. Volino, N. Magnenat-Thalmann, J. Shen, D. Thalmann. An involving system for simulating clothes on virtual actors. *IEEE Computer graphics and applications*. 16(5), pp. 42-51. Septembre 1996.
- (Ward et al., 1992) M.E. Ward, J.W. Ward, P.T. Macklem. Analysis of human chest wall motion using a two-compartment rib cage model. *Journal of Applied Physiology*. 72(4), pp 1338-1347. 1992.
- (Waters, 1987) K. Waters. A muscle model for animating three-dimensional facial expression. *Computer Graphics*, 21(4), pp 17-24. Juillet 1987. Siggraph 87.
- (Watt et Watt, 1992) A. Watt, M. Watt. *Advanced animation and rendering techniques, theory and practice*, Addison-Wesley. 1992.
- (West 1995) J.B. West. *Respiratory physiology, the essentials*, 5th edition, Baltimore, Williams & Wilkins, 1995.
- (Wilhelms, 1997) J. Wilhelms. Animals with anatomy. *IEEE Computer Graphics and Applications*, pp 22-30. Mai-Juin 1997 (voir aussi même auteur dans Siggraph 97).
- (Witkin et al., 1990) A. Witkin, M. Gleicher, W. Welch. Interactive dynamics. *Computer Graphics*, 24, pp 11-21. 1990. Proceedings 1990 of symposium on 3D interactive graphics.
- (Witkin et Welch, 1990) A. Witkin, W. Welch. Fast animation and control of nonrigid structures. *Computer Graphics*, 24(4), pp 243-252. 1990. Siggraph 90.
- (Witkin et al, 1995) A. Witkin, D. Barraf, M. Kass. An introduction to physically-based modeling. *Computer Graphics Siggraph'95 Course*. 1995.
- (Zienkiewicz et Taylor, 1991) O.C. Zienkiewicz, R.L. Taylor. *La méthode des éléments finis : formalisation de base et problèmes linéaires*. AFNOR technique. 1991.

Résumé

Le problème de la modélisation et de la simulation des mouvements du tronc pendant la respiration est étudié dans cette thèse. L'objectif est la représentation graphiquement réaliste de ces mouvements grâce à des modèles générateurs. Le découpage de ce document propose des étapes méthodologiques pour la modélisation.

Dans une première partie les principes mécaniques du phénomène de la respiration sont exposés. Le problème apparaît alors comme la modélisation et la simulation comportementale sous contraintes d'un objet complexe composés de régions ayant des propriétés différentes (élasticité, motricité, rigidité). Les modèles générateurs basés sur la physique se révèlent être les outils informatiques les plus aptes à atteindre notre objectif. Un état de l'art de ces méthodes et des problématiques liées est alors dressé.

La deuxième partie présente la construction du modèle informatique découpée en régions de propriétés spécifiques. On présente alors une nouvelle fonction d'élasticité utilisant l'expression d'une mémoire de forme locale. Comparée à un modèle classique masse-ressort, cette fonction d'élasticité prouve son efficacité. La modélisation des régions musculaires et solides est ensuite présentée. Puis, on montre la possibilité de contraindre ces régions afin de modéliser d'autres propriétés. On développe notamment une méthode de résolution directe permettant de vérifier des contraintes locales et globales sans utiliser d'algorithme itératif. La résolution de la contrainte d'incompressibilité illustre cette méthode de résolution ; les principes généraux de cette résolution sont dégagés permettant ainsi de généraliser son application.

La dernière partie de ce manuscrit fournit une validation qualitative du modèle développé par l'intermédiaire de différents exemples, dont celui de la simulation des mouvements du tronc pendant la respiration.

Abstract

The problem of modelling a mechanical biological system is studied in this thesis through the modelling and simulation of the trunk movements during respiration. The aim is the realistic graphical representation of these movements. The organisation of the present document suggests methodological steps for modelling.

In the first part, the mechanical principles of the respiratory phenomena are explained. Then, the problem appears as the modelling and behavioural simulation under constraints of a complex object composed of regions having different properties (elasticity, motor functioning and rigidity). Physically-based dynamic methods are suited to our aim. A state of the art of these methods and their problematic is drawn up.

The second part presents the making of the computer model by specific property regions. The geometric and dynamic based are proposed, the notion of force generated by an attractive point is defined. A new elastic function that uses this notion and the expression of local shape memory is presented. This function is then successfully compared to another elastic function. The modelling of muscle and solid regions is then explained. How to constrain the model is then study. We develop a new direct resolution method that allows verifying local and global constraints without any need for iterative algorithms. The incompressibility constraint resolution illustrates this resolution method, its principles are generalised to be used independently of our model.

The last part of this thesis gives a qualitative validation of our model by the use of some examples, including the simulation of trunk movements during respiration.