HAL
open science

Quelques contributions à la modélisation numérique de structures élancées pour l'informatique graphique

Romain Casati

▶ **To cite this version:**

Romain Casati. Quelques contributions à la modélisation numérique de structures élancées pour l'informatique graphique. Analyse numérique [math.NA]. Université Grenoble Alpes, 2015. Français. NNT : 2015GREAM053 . tel-01502808

HAL Id: tel-01502808

https://theses.hal.science/tel-01502808

Submitted on 6 Apr 2017

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.

UNIVERSITÉ DE GRENOBLE

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Mathématiques et Informatique**

Arrêté ministériel : 7 août 2006

Présentée par

Romain Casati

Thèse dirigée par **Bernard Brogliato**
et encadrée par **Florence Bertails-Descoubes**

préparée au sein du **Laboratoire Jean Kuntzmann**
et de l'école doctorale **EDMSTII**

Quelques contributions à la modélisation numérique de structures élancées pour l'informatique graphique

Thèse soutenue publiquement le **26 juin 2015**,
devant le jury composé de :

Éric Blayo
Professeur des universités, Université Joseph Fourier, Président
Loïc Barthe
Maître de conférences, Université Paul Sabatier, Rapporteur
Eitan Grinspun
Associate professor, Columbia University, Rapporteur
Basile Audoly
Directeur de recherche, CNRS – Paris 6, Examinateur
Evelyne Hubert
Chargée de recherche, Inria Sophia Antipolis – Méditerranée, Examinatrice
Bernard Brogliato
Directeur de recherche, Inria Rhône-Alpes, Directeur de thèse
Florence Bertails-Descoubes
Chargée de recherche, Inria Rhône-Alpes, Encadrante de thèse

Remerciements

TOUT d'abord, je remercie les personnes qui m'ont fait l'honneur d'être membre de ce jury. Loïc Barthe et Eitan Grinspun pour avoir accepté de rapporter cette thèse et pour s'être plongés dans mon travail. Merci à mes examinateurs, Basile Audoly et Evelyne Hubert ainsi qu'à Éric Blayo pour avoir accepté de présider ce jury.

Merci à Florence Bertails-Descoubes, mon encadrante de thèse, qui m'a permis de travailler sur ce sujet de recherche. Florence m'a toujours soutenu et a toujours pris le temps pour m'écouter malgré ses nombreuses contraintes. Elle a été un guide exemplaire et je peux dire que je suis toujours ressorti de son bureau plus motivé que je n'y étais entré. Merci pour ces cinq années de collaboration.

Je voudrais aussi remercier Bernard Brogliato pour m'avoir permis de réaliser cette thèse. Merci à Franck Hétroy pour m'avoir transmis sa passion de la recherche mais aussi pour son encadrement. Merci à Joëlle Thollot pour m'avoir pris sous son aile jusqu'à Siggraph en 2013, merci de m'avoir permis de ressembler à autre-chose qu'un flamant rose ! Merci aussi pour les discussions que nous avons eu à propos de mon avenir. Merci à Laurence Boissieux pour la qualité de son travail, ses idées, sa créativité et sa disponibilité. Merci à Jérôme Malick pour nos discussions, pour m'avoir soutenu dans mon passage de l'agrégation et pour m'avoir poussé à y croire.

Merci au personnel de l'Inria et en particulier aux assistantes d'équipe qui se sont succédées pour être au service de l'équipe Bipop. Merci aussi à Jean-François Scariot pour son sens du service et nos discussions toujours intéressantes.

Je remercie les doctorants de l'équipe Bipop pour la bonne ambiance à laquelle ils ont contribué. Merci à Sofia Zaourar-Michel, Gilles Daviet, Olivier Huber, Alejandro Blumentals, Mounia Haddouni, Alexandre Derouet-Jourdan et les autres.

Merci aussi à ma famille pour son soutien et son intérêt pour mon travail. Enfin, plus que tout le reste, merci à Hélène, sans qui tout serait bien terne, qui m'a toujours poussé à croire en ce que je faisais, qui a été d'un soutien indéfectible, merci de m'avoir supporté, merci pour ton amour.

Table des matières

Remerciements **3**

Table des matières **5**

Avant-propos **9**

1 Introduction générale **11**

Notations **15**

I Modélisation numérique de tiges élastiques **17**

2 Introduction **19**

3 État de l'art **23**

4 Le modèle des Super-Clothoïdes 3D **29**

4.1 Modèle de Cosserat, hypothèses de Kirchhoff 30
4.1.1 Le modèle de Cosserat 30
4.1.2 Les hypothèses de Kirchhoff 32
4.1.3 Problème de Darboux 33
4.1.4 Équations mécaniques d'une tige de Kirchhoff 35
4.1.5 Résolution des équations de Kirchhoff 39
4.2 Géométrie d'une Clothoïde 3D 40
4.2.1 Solution développable en série entière (DSE) 40
4.2.2 DSE pour la Clothoïde 3D, compensation catastrophique, remède 45
4.2.3 Raccordement des éléments de Clothoïde 3D 54
4.2.4 Cas d'un vecteur de courbures polynomial 55
4.3 Simulation dynamique d'une Super-Clothoïde 3D 58
4.3.1 Lagrangien, principe de moindre action, équation d'Euler–Lagrange 58
4.3.2 Application au calcul de la dynamique d'une Super-Clothoïde 3D 60
4.3.3 Calcul des quantités dynamiques par séries entières 64
4.3.4 Intégration en temps 67

5 Comparaisons et résultats **69**
5.1 Calcul de la géométrie : séries entières *versus* schémas d'intégration . . . 69
5.2 Comparaisons avec d'autres modèles dynamiques de tiges de Kirchhoff . . 73
5.2.1 Présentation des modèles 73
5.2.2 Description du protocole de comparaison 75
5.2.3 Comparaisons 76
5.3 Résultats 79
5.3.1 Plantes grimpantes volubiles 79
5.3.2 Simulation de chevelure 81
5.3.3 Animation de rubans 81

6 Conclusion de la première partie **85**

II Modélisation numérique inverse de coques élastiques en contact frottant **87**

7 Introduction **89**

8 État de l'art **95**

9 Choix d'un modèle de coques élastiques **99**
9.1 Difficultés de l'inversion d'un modèle élastique 100
9.2 Éléments de mécanique des coques élastiques 101
9.3 La simulation de coques élastiques en informatique graphique 106
9.4 L'inversion du modèle des *Discrete Shells* 108
9.5 Réflexions sur l'élaboration d'un modèle de coques élastiques destiné à l'inversion 108

10 Inversion de coques en contact frottant **115**
10.1 Cas sans contact 116
10.1.1 Première approche : minimisation de l'erreur d'équation d'état . . 116
10.1.2 Fonction de drapé, formulation aux moindres carrés 117
10.1.3 Procédure de drapé, appels cohérents 121
10.1.4 Forme au repos initiale 123
10.1.5 Algorithme complet 123
10.2 Cas des points fixés 124
10.3 Intermède : initiation au contact frottant 125
10.3.1 Le modèle de frottement de Coulomb 125
10.3.2 Outils d'analyse convexe 127
10.4 Prise en compte du contact frottant 129
10.4.1 Le drapé comme un problème d'optimisation 131
10.4.2 Il faut sauver l'adjoint 133
10.4.3 Discussion 133

Compléments **135**
10.A Modification de l'adjoint pour l'ajout de contraintes 135
10.A.1 Cas des points fixés 135
10.A.2 Cas du contact frottant 136

11 Résultats **141**
11.1 Contact bilatéral *versus* contact unilatéral avec frottement 141
11.2 Validation 141

12 Conclusion de la seconde partie **149**

13 Perspectives **153**

Annexes **157**

A Forme explicite et forme fermée **157**

B Rappels sur les séries entières **159**

C Théorème des fonctions implicites **161**

D Quelques formules et résultats sur les rotations **163**
D.1 Matrice de produit vectoriel 163
D.2 Rotations 164
D.3 Géométrie différentielle sur $\mathcal{SO}(3)$ 165

Bibliographie **167**

Résumé – Abstract **178**

Avant-propos

Pour faciliter la lecture de ce document, les équations, définitions, théorèmes, algorithmes, remarques, *etc.*, partagent la même numérotation et celle-ci se fait par section. Par exemple, dans la section 4.2_{40} du chapitre 4_{29}, le premier élément est numéroté $4.2.1_{40}$; c'est un théorème dont la première équation est numérotée $4.2.2_{40}$, la suivante $4.2.3_{40}$ et ainsi de suite. Il se peut que certains éléments, dits « flottants », comme les figures ou les algorithmes, qui sont disposés librement par LaTeX, ne respectent pas parfaitement la continuité de la numérotation. Dans tous les cas, soit l'élément auquel il est fait référence se trouve sur une autre page et elle est spécifiée en indice, comme dans « $4.2.1_{40}$ », indiquant que le théorème numéroté 4.2.1 se trouve page 40, soit il se trouve sur la même page et ceci est indiqué par l'absence d'indice, comme dans « 4.2.1 ».

Les notations utilisées dans ce document sont listées page 15.

Enfin, certaines annexes, qui prennent plutôt la forme de compléments, contiennent des contributions importantes mais dont le détail, au moment où il en est question, risquerait de perturber l'exposition générale des travaux ; ces compléments sont alors ajoutés à la fin du chapitre correspondant. D'autres annexes, en fin de manuscrit, reprennent des résultats classiques, accompagnés de quelques commentaires, sous un angle et avec des notations adaptées à l'utilisation qui en est faite dans le corps de la thèse.

Chapitre 1

Introduction générale

Il est intéressant d'observer qu'une grande partie des objets déformables qui nous entourent sont caractérisés par une forme élancée : soit filiforme, comme les cheveux, les plantes, les fils (voir la figure 1.0.1) ; soit surfacique, comme le papier, les feuilles d'arbres, les vêtements ou la plupart des emballages (voir la figure $1.0.2_{12}$).

(a) (b) (c)

Figure 1.0.1 : Des exemples d'objets filiformes autour de nous. L'image (a) est tirée de (Pickover, 2012), (b) de (Casati et Bertails-Descoubes, 2013) et (c) de (Daviet *et al.*, 2011).

Simuler (numériquement) la mécanique de telles structures présente alors un intérêt certain : cela permet de prédire leur comportement dynamique, leur forme statique ou encore les efforts qu'elles subissent. Cependant, pour pouvoir réaliser correctement ces simulations, plusieurs problèmes se posent. Les modèles (mécaniques, numériques) utilisés doivent être adaptés aux phénomènes que l'on souhaite reproduire, ainsi qu'à l'échelle d'étude ; le modèle mécanique choisi doit pouvoir être traité numériquement, c'est-à-dire que les équations qui le constituent doivent pouvoir être résolues soit de manière exacte, soit de manière approchée par des méthodes compatibles avec l'architecture utilisée[1] ; enfin, il est nécessaire de connaître les paramètres du modèle qui permettront

[1]Par exemple, si l'on travaille en arithmétique flottante, les erreurs d'arrondi peuvent devenir pro-

(a) (b) (c)

FIGURE 1.0.2 : Quelques objets surfaciques déformables. Le maillage de la figure (b) est issue d'une capture provenant de (White *et al.*, 2007) et la figure (c) est tirée de (Grinspun *et al.*, 2003)

de reproduire l'instance du phénomène souhaitée. Dans cette thèse nous abordons ces trois points, dans le cadre de la simulation de structures élancées.

Dans la première partie, nous décrivons une nouvelle méthode de discrétisation spatiale des équations de Kirchhoff dynamiques, régissant le comportement des tiges élastiques inextensibles et sans cisaillement. L'approche consiste à découper la tige en éléments de courbures et de torsion affines avec l'abscisse curviligne, une extension des clothoïdes à l'espace. La difficulté majeure tient au fait que le problème géométrique ne possède pas de solution explicite ; nous montrons cependant qu'une utilisation des séries entières avec une attention particulière pour éviter les problèmes d'arithmétique flottante permet non seulement de résoudre efficacement et précisément le problème géométrique mais aussi d'intégrer la dynamique de manière performante. Nous comparons soigneusement la précision géométrique ainsi que l'efficacité de notre approche avec des modèles de référence et montrons la pertinence d'une telle contribution.

Dans la seconde partie, nous nous intéressons à la conception d'une méthode de calcul automatique des paramètres géométriques d'un modèle de surface déformable (coque élastique). Plus précisément, on trouve la forme au repos (sans force appliquée) que doit avoir une coque élastique, connaissant sa forme à l'équilibre sous la gravité et en présence de contact frottant, et connaissant les paramètres physiques du matériau. Un tel problème est dit *inverse* puisqu'il procède dans le sens opposé à celui de la simulation. Nous apportons aussi des éléments de réflexion pour l'élaboration d'un modèle qui pourrait faciliter le processus d'inversion.

Les deux parties sont dans une large mesure indépendantes. Cependant, nous avons à plusieurs reprises dans la seconde partie, mis en évidence les connexions qui existent avec la première. D'autre part, les perspectives que nous donnons en fin de document marient ces deux parties.

---

blématiques.

Dans cette thèse, nous nous sommes intéressés essentiellement aux applications qui concernent l'informatique graphique, dans lesquelles richesse visuelle, robustesse et efficacité des simulateurs, ainsi que le contrôle de l'utilisateur, jouent un rôle primordial. Par ailleurs, la complexité et la taille des instances à traiter (des centaines ou milliers de fibres en contact frottant, des vêtements soumis à de grands déplacements et en contact frottant avec le corps du personnage, *etc.*) en font de réels défis, même pour des méthodes ayant fait leurs preuves dans un contexte différent. D'autre part, l'intérêt croissant des physiciens pour des modèles numériques compacts et performants nous laisse espérer que l'applicabilité des méthodes proposées pourra dépasser le champ de l'informatique graphique.

Notations

$\varnothing$	Ensemble vide.
$\mathbb{N}$	Ensemble des entiers naturels.
$\mathbb{Z}$	Ensemble des entiers relatifs.
$\mathbb{R}$	Ensemble des nombres réels.
$\mathbb{R}^*$	Ensemble des nombres réels privé de l'élément nul.
$\mathbb{R}_+$	Ensemble des nombres réels positifs ou nul.
$\mathbb{R}_-$	Ensemble des nombres réels négatifs ou nul.
$[a\,;b]$	Ensemble des nombres réels compris (au sens large) entre les réels a et b.
$[\![a\,;b]\!]$	Segment contenant les entiers compris entre les réels a et b, $[a\,;b]\cap\mathbb{Z}$.
$\lfloor x\rfloor$	Partie entière par défaut de $x\in\mathbb{R}$, *i.e.* plus grand entier $n\in\mathbb{Z}$ tel que $n\leqslant x<n+1$.
$\mathbb{C}$	Ensemble des nombres complexes.
$\mathbb{R}^n$	$\underbrace{\mathbb{R}\times\mathbb{R}\times\cdots\times\mathbb{R}}_{n\text{ fois}}$, c'est l'ensemble des n-uplets de réels.
$\mathring{A}$	Intérieur de $A\subset\mathbb{R}^n$, c'est le plus grand ouvert contenu dans A.
$\overline{A}$	Adhérence de $A\subset\mathbb{R}^n$, c'est le plus petit fermé contenant A.
∂A	Frontière de $A\subset\mathbb{R}^n$, c'est l'adhérence privée de l'intérieur, $\overline{A}\backslash\mathring{A}$.
$\mathscr{C}^n$	Ensemble des applications n fois dérivables et dont la dérivée n-ième est continue, $n=\infty$ éventuellement. Les ensembles de départ et d'arrivée sont soit précisés entre parenthèses, soit déterminés par le contexte.
G^n	Ensemble des courbes pour lesquelles il existe une paramétrisation de classe $\mathscr{C}^n$, $n=\infty$ éventuellement.
$u\times v$	Produit vectoriel des vecteurs u et v de $\mathbb{R}^3$ (nous utilisons la notation anglo-saxone).
o, $\mathcal{O}$, Θ	Relations de comparaison usuelles, $f=\Theta(g)$ signifie $f=\mathcal{O}(g)$ et $g=\mathcal{O}(f)$.
$Df(x)$	Différentielle (au sens de Fréchet) de $f:\mathbb{R}^n\longrightarrow\mathbb{R}^p$ en $x\in\mathbb{R}^n$, aussi appelée *application linéaire tangente*, c'est une application linéaire de $\mathbb{R}^n$ dans $\mathbb{R}^p$. On peut parfois préciser en indice la variable de différentiation comme dans D_xf.
$D^2f(x)$	Différentielle seconde de $f:\mathbb{R}^n\longrightarrow\mathbb{R}^p$ en $x\in\mathbb{R}^n$, c'est une application bilinéaire symétrique de $\mathbb{R}^n\times\mathbb{R}^n$ dans $\mathbb{R}^p$. On peut parfois préciser en indice les variables de différentiation comme dans $D^2_{x,y}f$.

$\nabla f(x)$	Gradient de $f : \mathbb{R}^n \longrightarrow \mathbb{R}$ au point $x \in \mathbb{R}^n$, c'est une forme linéaire de $\mathbb{R}^n$ dans $\mathbb{R}$ identifiée à un vecteur de $\mathbb{R}^n$. On peut parfois préciser en indice la variable de différentiation comme dans $\nabla_x f$.
$\dfrac{\partial f}{\partial x_i}$	Dérivée partielle de $f : \mathbb{R}^n \longrightarrow \mathbb{R}^p$ par rapport à la variable x_i, comme f, c'est une application de $\mathbb{R}^n$ dans $\mathbb{R}^p$.
$\dfrac{\partial^2 f}{\partial x_i \partial x_j}$	Dérivée partielle seconde de $f : \mathbb{R}^n \longrightarrow \mathbb{R}$ par rapport aux variables x_i et x_j, comme f, c'est une application de $\mathbb{R}^n$ dans $\mathbb{R}^p$.
$\langle \cdot \,;\, \cdot \rangle$	Produit scalaire, généralement de $\mathbb{R}^n \times \mathbb{R}^n$ dans $\mathbb{R}$ mais peut être précisé suivant le contexte.
$\lVert \cdot \rVert$	Norme dépendante du contexte (souvent associée au produit scalaire ou norme sup).
$E^\perp$	Ensemble des vecteurs de $\mathbb{R}^n$ qui sont orthogonaux à tous les vecteurs de $E \subset \mathbb{R}^n$.
$\mathcal{S}^{n-1}(\mathbb{R})$	Sphère unité de $\mathbb{R}^n$ pour la norme du contexte.
$\mathrm{Im}(f)$	Image de l'application linéaire f.
$\mathrm{Ker}(f)$	Noyau de l'application linéaire f (image réciproque de $\{0\}$).
$\dim(F)$	Dimenssion du sous-espace vecotiel $F \subset \mathbb{R}^n$.
$\mathcal{M}_n(\mathbb{R})$	Ensemble des matrices carrées de taille $n \times n$ à coefficients dans $\mathbb{R}$.
I	Matrice identité de $\mathcal{M}_n(\mathbb{R})$ (élément neutre pour le produit).
$A^\top$	Transposée de la matrice $A \in \mathcal{M}_n(\mathbb{R})$.
$\mathcal{S}_n(\mathbb{R})$	Ensemble des matrices symétriques de $\mathcal{M}_n(\mathbb{R})$, *i.e.* vérifiant $A^\top = A$.
$\mathcal{A}_n(\mathbb{R})$	Ensemble des matrices antisymétriques de $\mathcal{M}_n(\mathbb{R})$, *i.e.* vérifiant $A^\top = -A$.
$\mathrm{tr}(A)$	Trace de la matrice $A \in \mathcal{M}_n(\mathbb{R})$ (somme des éléments diagonaux).
$\det(A)$	Déterminant de la matrice $A \in \mathcal{M}_n(\mathbb{R})$.
$\mathcal{O}(n)$	Groupe des matrices orthogonales de $\mathcal{M}_n(\mathbb{R})$, *i.e.* vérifiant $A^\top A = I$.
$\mathcal{SO}(n)$	Sous-groupe de $\mathcal{O}(n)$ dont les matrices ont un déterminant égal à 1.
$\mathcal{S}_n^{++}(\mathbb{R})$	Ensemble des matrices symétriques définies positives de $\mathcal{M}_n(\mathbb{R})$, *i.e* vérifiant $(\langle Ax\,;x\rangle \geqslant 0$ et $(\langle Ax\,;x\rangle = 0 \implies x = 0))$ pour tout vecteur x de $\mathbb{R}^n$. On l'identifie à l'ensemble des formes bilinéaires symétriques définies positives de $\mathbb{R}^n \times \mathbb{R}^n \longrightarrow \mathbb{R}$, lui-même identifié à l'ensemble des formes quadratiques définies positives de $\mathbb{R}^n$ dans $\mathbb{R}$.
$\langle \cdot \,;\, \cdot \rangle_M$	Produit scalaire associé à la matrice M de $\mathcal{S}_n^{++}(\mathbb{R})$, pour $u, v \in \mathbb{R}^n$, $\langle u\,;v\rangle_M = \langle Mu\,;v\rangle$.
$\lVert \cdot \rVert_M$	Norme associée au produit scalaire $\langle \cdot \,;\, \cdot \rangle_M$.
$T_x M$	Espace tangent à la sous-variété M en x.

Première partie

Modélisation numérique de tiges élastiques

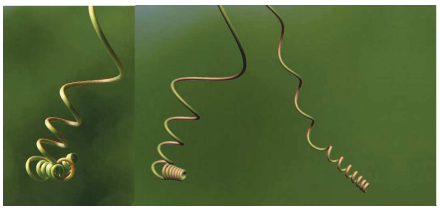
Chapitre 2

Introduction

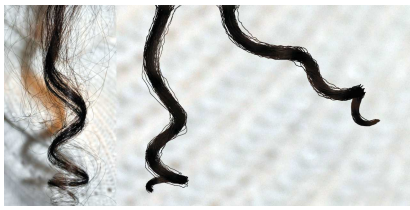
L'UN des principaux intérêts de l'informatique graphique est la création de formes et de mouvements virtuels qui synthétisent la complexité visuelle du monde qui nous entoure. Les objets longilignes se retrouvent dans les plantes (tiges), chez les humains et animaux (cheveux et fourrures) ou dans les objets manufacturés (fils, câbles, rubans). Quelques-uns de ces objets sont représentés figure $2.0.1_{20}$. Ils sont caractérisés par un comportement mécanique complexe donnant lieu à des formes lisses et courbées, et participent ainsi à une grande richesse visuelle. Notre but ici est d'aboutir à un modèle numérique pour simuler de manière robuste et efficace la mécanique des tiges, en considérant en particulier celles qui sont les plus courbées.

Ces tiges flexibles, que l'on peut supposer inextensibles et sans cisaillement, sont bien décrites par la théorie non-linéaire de Kirchhoff (Dill, 1992). Cependant, les équations du mouvement sont réputées comme étant difficiles à traiter numériquement, notamment par la présence de dérivées d'ordre 4 en espace. L'inextensibilité et les forces élastiques doivent en particulier être traitées avec soin car elles constituent la plus grande source d'instabilité numérique.

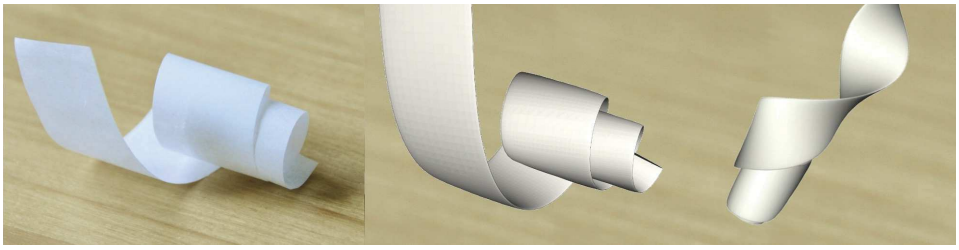
La plupart des méthodes numériques existantes reposent sur une discrétisation nodale de la tige qui a l'avantage de mener à des équations creuses, mais qui nécessite cependant beaucoup d'éléments pour représenter des géométries fortement courbées. D'autre part, ces méthodes ne permettent pas de garantir l'inextensibilité explicitement et cette dernière doit alors être contrainte au risque de compromettre l'efficacité ou la stabilité du système. Ces aspects seront détaillés au chapitre 3_{23}. À l'inverse, nous cherchons une paramétrisation réduite de haut degré incorporant de manière compacte toute la cinématique de la tige. Dans cette optique, le modèle des Super-Hélices, qui se compose d'éléments d'hélices circulaires parfaitement inextensibles, et pour lequel les forces élastiques sont linéaires par rapport aux degrés de liberté, a été la première approche introduite dans ce sens (Bertails *et al.*, 2006). Cependant, ce modèle possède un ordre de continuité faible (G^1) dû à la discontinuité des courbures. Dans la nature, la plupart des tiges possèdent une courbure continue et sont bien approchées par des éléments de



(a)



(b)



(c)

FIGURE 2.0.1 : Des objets courants, filiformes et courbés. Une vrille de vigne (a), une mèche de cheveux (b) et un ruban de papier (c). À droite de chacun des objets, deux images d'une simulation utilisant notre modèle en courbure affine par morceaux. Remarquer la ressemblance des formes même à faible résolution : 4 éléments pour (a), 2 pour (b) et 1 pour (c). Photo (a) à gauche, © Jon Sullivan, pdphoto.org.

courbures affines par morceaux (voir la figure 2.0.1). Ces observations nous amènent à proposer, au chapitre 4_{29}, une primitive dont la courbe moyenne est une Clothoïde 3D ou spirale d'Euler 3D — une courbe dont la courbure et la torsion varient linéairement[1] (Harary et Tal, 2012). Le modèle résultant du raccordement G^2 de tels éléments est celui que nous appellerons Super-Clothoïdes 3D ; il reste parfaitement inextensible.

La difficulté majeure qui se présente alors, et qui n'est pas rencontrée par les modèles de plus faible degré, est celle de l'évaluation de la géométrie (et par conséquent de toute la cinématique de la tige). En effet, l'expression de forme fermée (voir l'annexe A_{157} pour une définition) sur laquelle reposait le modèle des Super-Hélices n'existe plus mais il est pourtant nécessaire que cette géométrie soit évaluée précisément. Dans le cas de la dimension 2, les équations prennent la forme d'un problème, plus simple, de quadrature (Bertails-Descoubes, 2012). Ceci permet l'intégration rapide de la dynamique pour la dimension 2, mais n'est pas applicable à la dimension 3.

Notre approche est inspirée des méthodes de calcul symbolique dont l'objectif est d'obtenir des résultats avec une très grande précision et une majoration de l'erreur commise (Mezzarobba, 2011). Si la précision est aussi une de nos préoccupations, l'efficacité

[1] Plus précisément, ce sont des courbures et torsion *matérielles* qui sont linéaires dans notre modèle mais les courbes ainsi décrites sont plus générales puisqu'il suffit d'imposer que la première courbure soit nulle pour retrouver les courbes de (Harary et Tal, 2012).

est cruciale pour permettre le calcul rapide des quantités coûteuses comme celles qui interviennent dans le calcul de la dynamique. Nous nous attachons donc à développer une méthode qui utilise uniquement l'arithmétique flottante standard (simple ou double précision).

Le cœur de notre contribution est une nouvelle manière de résoudre la cinématique et la dynamique des tiges en exploitant la structure des équations, grâce à des séries entières, tout en prenant soin d'éviter les problèmes numériques liés à l'arithmétique flottante. Au chapitre 5_{69}, nous montrons d'une part que notre méthode permet d'atteindre la précision machine beaucoup plus rapidement que les intégrateurs numériques classiques ; et d'autre part que notre dicrétisation spatiale converge beaucoup plus rapidement vers la solution du problème continu que les autres modèles de tiges. Nos résultats illustrent la richesse des mouvements générés à des temps de calcul compétitifs.

Chapitre 3

État de l'art

Les tiges flexibles, ces structures déformables élancées dans une direction, ont été étudiées depuis plus d'un siècle dans plusieurs domaines scientifiques. Les applications sont en effet nombreuses et à des échelles très différentes, allant de la pose de câbles sous-marins (Goyal *et al.*, 2008) à l'étude de l'enroulement des brins d'ADN (Benham et Mielke, 2005), et en passant par la croissance des plantes grimpantes (Goriely et Neukirch, 2006), la modélisation de chevelures (Ward *et al.*, 2006), ou encore la simulation de fils chirurgicaux (Chentanez *et al.*, 2009). Voir la figure 3.0.1$_{24}$ pour une illustration de cette variété d'applications.

Théories des tiges élastiques minces

On distingue plusieurs modèles mécaniques de tiges déformables en fonction du type de déformation qu'elles ont à subir. Ici nous nous intéressons à la modélisation des cheveux bouclés, des rubans enroulés, des plantes ou des câbles vrillés. Tous ces exemples subissent des déformations *élastiques* (*i.e.*, réversibles) qui sont dominées par la *flexion* et la *torsion* (modérées) alors que leur *extensibilité* et leur *cisaillement* peuvent être négligés. C'est pourquoi nous considérons un modèle *inextensible*, sans cisaillement et dont les déformations sont petites de sorte que la loi de comportement élastique est linéaire. Pour rendre compte des non-linéarités caractéristiques des phénomènes que nous avons mentionnés plus haut, les grands déplacements doivent être autorisés. Les équations régissant le mouvement d'un tel solide ont été introduites par Kirchhoff et Clebsch dans leur théorie des tiges élastiques minces en grands déplacements (Dill, 1992). Plus tard, les frères Cosserat et Cosserat (1909) ont proposé une représentation mathématique intuitive dans laquelle la tige est modélisée par une courbe (la *courbe moyenne*) et une *base matérielle* décrivant les déformations de la section de la tige et tournant le long de la courbe autour du *vecteur de Darboux*. On trouvera dans (Audoly et Pomeau, 2010) une description détaillée de cette théorie, dérivée de l'élasticité tridimensionnelle.

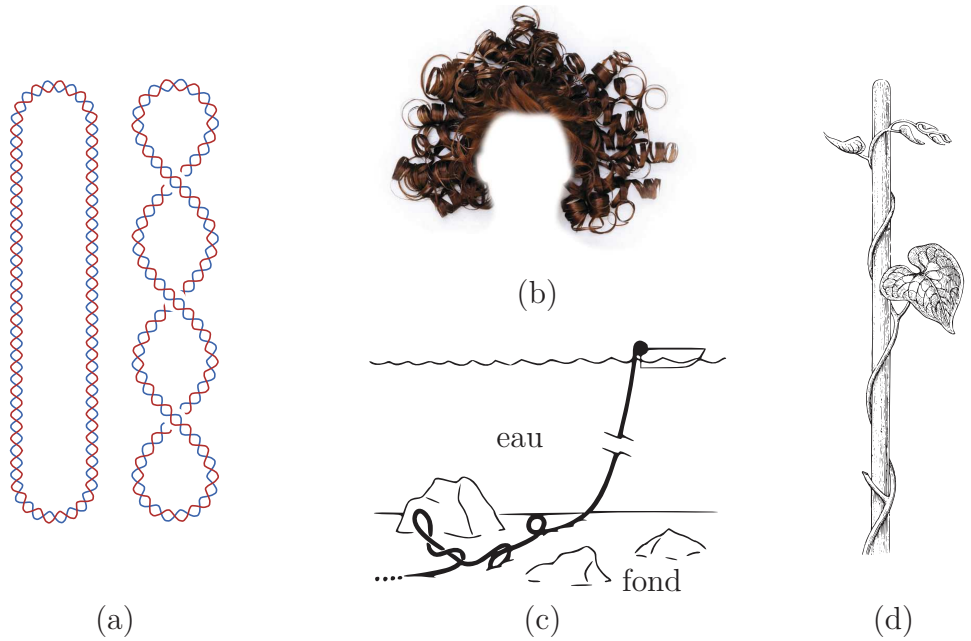


FIGURE 3.0.1 : Différentes tiges à différentes échelles : des brins d'ADN droit et enroulé (a), des cheveux (b), un câble sous-marin (c) et une plante grimpante volubile (d) (Sachs *et al.*, 1875).

Solution des équations

Il n'existe pas, dans le cas général, de solution explicite aux équations de Kirchhoff ; des techniques permettant d'approcher ces solutions doivent alors être développées. Parmi les méthodes les plus connues pour discrétiser les équations de Kirchhoff en temps et en espace, citons les différences finies et les éléments finis. Même si les méthodes à base de différences finies sont souvent simples à mettre en œuvre, la prise en compte des conditions aux bords n'est pas toujours évidente dans le cas des tiges et nécessite souvent l'utilisation d'une *méthode de tir*[1]. De plus, le caractère raide des équations de Kirchhoff et la présence de dérivées d'ordre 4 (en espace) imposent l'utilisation de petits pas d'intégration ou d'intégrateurs implicites adaptés (Goyal *et al.*, 2008). Les méthodes à base d'éléments finis permettent de transformer les équations aux dérivées partielles en une équation différentielle ordinaire en « éliminant » la variable d'espace grâce à des éléments bien choisis. Parmi ces méthodes, on trouve l'approche connue sous le nom des *tiges géométriquement exactes* (Simo et Vu-Quoc, 1986) qui, dans le cadre plus général d'une tige extensible et soumise au cisaillement, discrétise le champ de déplacement et celui des rotations par des fonctions de forme interpolantes. Le problème majeur de cette approche, qui a donné lieu à beaucoup de travaux, est celui de l'utilisation d'éléments interpolants

[1]Les méthodes de tir s'appliquent à la résolution des problèmes aux limites où les conditions (inconnues) à un bord sont approchées itérativement pour satisfaire les conditions au bord opposé, par des résolutions successives du problème de Cauchy correspondant.

préservant l'*objectivité*, c'est-à-dire l'invariance du tenseur métrique aux transformations rigides Crisfield et Jelenić (1999). D'autre part, sauf à utiliser des constantes de raideur importantes et obtenir un système raide, il n'est pas possible d'appliquer directement cette méthode au cas des tiges inextensibles et sans cisaillement.

En informatique graphique, le premier à avoir introduit le modèle de Kirchhoff à la communauté a été Pai (2002). Son approche, limitée au problème statique, se base sur une discrétisation spatiale par différences finies. Dans la suite, plusieurs approches se sont attaquées au problème dynamique, grâce à des techniques plus avancées basées sur des formulations variationnelles ou des concepts de géométrie différentielle discrète. Parmi les méthodes proposées, on peut distinguer deux grandes classes de modèles (Baraff, 1996) ; d'une part, les modèles en *coordonnées réduites* (ou *modèles réduits*), basés sur une paramétrisation minimale du système, et d'autre part, les modèles en *coordonnées maximales* (ou *modèles nodaux*) dont les degrés de liberté sont des points échantillonnés sur la courbe moyenne assortis de contraintes cinématiques. Ces derniers sont caractérisés par une matrice de masse creuse qui permet souvent une intégration en temps efficace même pour un grand nombre de degrés de liberté. À l'inverse, les modèles réduits héritent la plupart du temps d'une matrice de masse dense mais sont souvent exempts de contrainte cinématique. Dans cette seconde catégorie, deux travaux ont permis de décrire la cinématique exacte des tiges et de manière parfaitement inextensible. Il s'agit de l'approche des solides rigides articulés (Hadap et Magnenat-Thalmann, 2001; Hadap, 2006) paramétrés par des articulations angulaires ; et les Super-Hélices (Bertails *et al.*, 2006) paramétrées par des courbures et torsion constantes par morceaux. Dans la catégorie des modèles nodaux, on trouve le modèle CoRdE (Spillmann et Teschner, 2007) où les degrés de liberté sont les positions et les orientations de la section de la tige. Les orientations sont couplées aux positions par une contrainte souple (pénalité) et l'extensibilité est limitée grâce à un terme élastique raide. On trouve aussi le modèle des *Discrete Elastic Rods* (Bergou *et al.*, 2008) qui, en se fondant sur le repère de Bishop, utilise des angles associés aux nœuds de la discrétisation pour paramétrer astucieusement la base matérielle. Ceci garantit notamment que la base matérielle reste par construction *adaptée* à la courbe moyenne, c'est-à-dire que l'un de ses vecteurs reste aligné avec la tangente à la courbe. L'inextensibilité est traitée explicitement, en utilisant soit une méthode de projection rapide (Bergou *et al.*, 2008; Goldenthal *et al.*, 2007), soit un terme élastique raide (Bergou *et al.*, 2010). De plus, pour assurer une certaine stabilité, c'est un schéma implicite, résolu avec un algorithme de type Newton–Raphson, qui est utilisé afin de maîtriser les termes raides de flexion/torsion et d'élongation/compression (Bergou *et al.*, 2010).

Modèles réduits basés sur les courbures et la torsion

L'avantage de considérer les courbures et la torsion comme degrés de liberté, au lieu des positions, est que la cinématique de la tige est alors préservée de manière exacte (en particulier l'inextensibilité), sans redondance dans la représentation et donc sans contrainte supplémentaire. Un tel modèle réduit et de haut degré bénéficie aussi d'une

implicitation « à moindre frais » des forces élastiques, puisque ces dernières dépendent linéairement des courbures et de la torsion. Enfin, si les modèles nodaux ont bénéficié de nombreuses méthodes spécifiques pour la réponse à la collision (comme « l'altération de position » (Baraff et Witkin, 1998)), des solveurs de contact frottant performants se sont révélés être facilement utilisables avec ce genre de modèles réduits (Daviet *et al.*, 2011).

Comme les méthodes d'éléments finis dont il est question plus haut, les Super-Hélices reposent sur une formulation faible des équations de Kirchhoff. La différence principale réside dans le fait que la discrétisation n'opère pas sur le champ des déplacements et des rotations, mais sur celui des courbures et de la torsion. Courbures et torsion sont approchées par des fonctions constantes par morceaux et la courbe moyenne résultante est alors constituée d'hélices circulaires. Cette formulation a l'avantage non seulement de tenir compte exactement de l'inextensibilité mais aussi de ne pas souffrir du problème de non-objectivité dont il est question plus haut. Le prix à payer est le caractère dense de la matrice de masse, mais ceci est nuancé par le petit nombre d'éléments (entre 5 et 10) nécessaire pour capturer des mouvements et des formes visuellement riches.

Cependant, comme illustré figures 3.0.2 et $3.0.4_{28}$, les jonctions seulement $\mathscr{C}^1$ des Super-Hélices restent visibles et disgracieuses, surtout à faible résolution. Comme dans

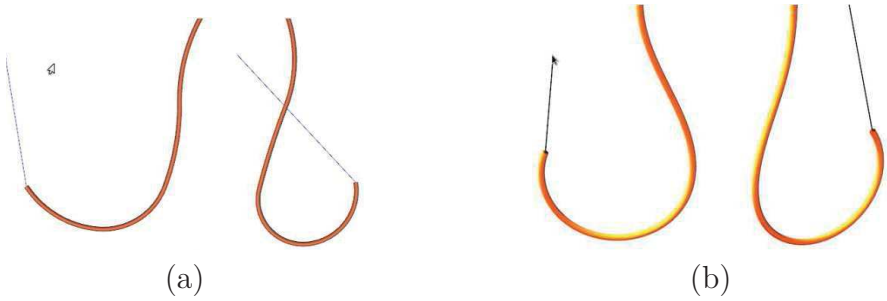


FIGURE 3.0.2 : Comparaison esthétique entre (a) une Super-Hélice et (b) notre modèle de Super Clothoïde 3D. Les jonctions entres les hélices sont particulièrement visibles (5 éléments) alors que notre modèle possède une forme esthétique même à faible résolution (2 éléments).

le cas de la dimension 2 (Bertails-Descoubes, 2012), ceci peut-être corrigé en imposant une continuité $\mathscr{C}^2$ grâce à l'utilisation de clothoïdes par exemple (voir la figure $3.0.4_{28}$).

Utilisation des clothoïdes

Les clothoïdes, aussi appelées spirales de Cornu ou d'Euler, sont des courbes planes le long desquelles la courbure varie de manière affine. La figure $3.0.3_{27}$ donne le tracé d'une clothoïde dont la courbure vérifie $\kappa(s) = \frac{s}{2}$. Les clothoïdes permettent des jonctions continues en courbure (voir la figure $3.0.4_{28}$) ce qui justifie principalement leur utilisation dans la conception de routes, ou de la trajectoires de robots (Delingette *et al.*, 1991;

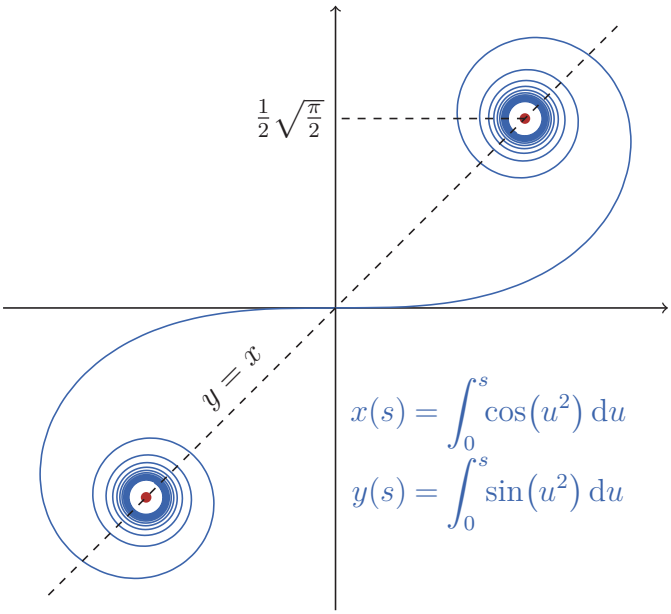


FIGURE 3.0.3 : La spirale de Cornu dont les équations sont données par les intégrales de Fresnel.

Fraichard et Scheuer, 2004), car elles garantissent la continuité de l'accélération le long de la trajectoire. Leurs qualités esthétiques, dues à la minimisation de la variation en courbure $\int \kappa'^2$, en font également de très bonnes candidates à la création de polices d'écriture (Levien, 2009), la complétion de formes (Kimia *et al.*, 2003; Harary et Tal, 2012), la conception géométrique assistée par ordinateur (McCrae et Singh, 2008; Baran *et al.*, 2010) ou encore la simulation physique de courbes 2D (Bertails-Descoubes, 2012). Leur évaluation repose sur les intégrales de Fresnel ou sur des méthodes de quadrature puisqu'elles ne possèdent pas d'expression de *forme fermée* (voir l'annexe A_{157} pour une définition). Leur extension au cas d'une courbure polynomiale semble remonter à Dillen (1990), qui leur donne le nom de *spirales polynomiales*. Elles aussi sont utilisées pour la génération de trajectoire de robots (Kelly et Nagy, 2003; McNaughton *et al.*, 2011) ou la conception de polices de caractères (Levien, 2009). Comme pour les clothoïdes, ce sont le plus souvent des méthodes de quadrature qui sont utilisées pour leur évaluation (Kelly et Nagy, 2003; McNaughton *et al.*, 2011) même si on note dans (Levien, 2009) l'utilisation de développements limités grâce à la génération de code. Enfin, les clothoïdes ont déjà été étendues à la dimension 3 (Harary et Tal, 2012) pour la complétion de formes : dans cette extension, comme on s'y attend, courbure et torsion géométriques sont des fonctions affines de l'abscisse curviligne. Cependant, contrairement à la dimension 2, l'évaluation de la géométrie des courbes résultantes est plus complexe et repose sur la résolution d'une équation différentielle linéaire, intégrée numériquement. Dans la suite, nous considérons une classe de courbes 3D plus large, en étendant la définition de Harary et Tal (2012) pour une clothoïde 3D.

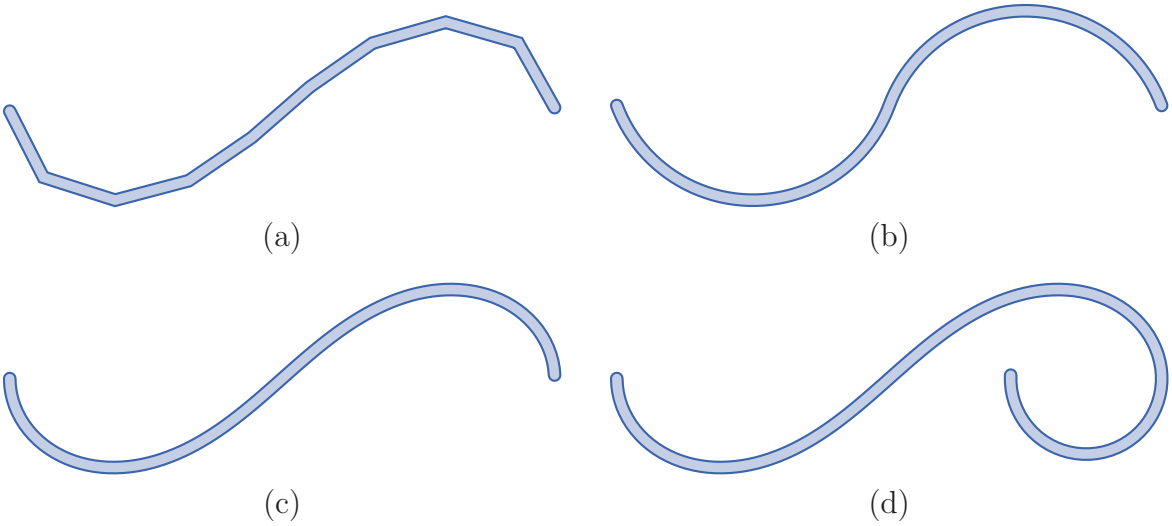


FIGURE 3.0.4 : Comparaison des degrés de continuité en dimension 2. Les modèles nodaux ou à base de segments (Bergou *et al.*, 2008; Hadap, 2006) possèdent des discontinuités tangentielles (a) ; les modèles à base d'arcs de cercles (ou d'hélices) (Bertails *et al.*, 2006) sont discontinus en courbure (b), et cette discontinuité est particulièrement visible et déplaisante visuellement (au milieu de la courbe) ; la clothoide est au contraire une primitive très esthétique (c) et peut être reliée à un arc de cercle (par exemple) de manière $\mathscr{C}^2$ sans discontinuité de la courbure (d).

Chapitre 4

Le modèle des Super-Clothoïdes 3D

Dans ce chapitre nous présentons notre modèle discret de simulation de tiges mécaniques, appelé *Super-Clothoïdes 3D*. La première section est consacrée à la motivation et à l'exposition des équations du modèle dynamique continu des *tiges de Kirchhoff* et nous montrons que leur résolution est un problème difficile nécessitant des techniques d'approximation robustes.

La discrétisation spatiale de ces équations par des éléments en coordonnées nodales (Bergou *et al.*, 2008) ou en coordonnées réduites en hélices circulaires (Bertails *et al.*, 2006) ont permis d'approcher numériquement les solutions des équations dynamiques des tiges de Kirchhoff. Afin de réduire le nombre d'éléments nécessaires à une bonne approximation et espérer ainsi, peut-être, améliorer l'efficacité en temps de calcul, nous avons considéré des éléments de degré encore plus élevé que des hélices.

Dans la seconde section de ce chapitre, nous présentons notre discrétisation géométrique par des éléments de courbures et torsion qui varient de manière affine avec l'abscisse curviligne, une généralisation des clothoïdes du plan que nous appelons les *Clothoïdes 3D*. Le modèle présenté est à la fois de haut degré, et en coordonnées réduites. Nous montrons aussi que ce choix de coordonnées permet d'améliorer la continuité de la géométrie sous-jacente du modèle, dont la courbe moyenne est de classe $\mathscr{C}^2$, rendant ainsi inutile l'utilisation d'une géométrie lisse supplémentaire (contrairement au cas des modèles de faible degré) pour la visualisation ou la détection de collision.

Ensuite, nous montrons comment notre méthode de calcul de la géométrie permet une évaluation efficace des quantités nécessaires à la simulation numérique de ce modèle dynamique que nous appelons les *Super-Clothoïdes 3D*.

Enfin, afin de mesurer l'apport de notre modèle sur l'état de l'art, nous le comparons géométriquement et dynamiquement avec les deux modèles discrets de tiges de Kirchhoff dynamiques introduits en informatique graphique, les *Discrete Elastic Rods* (Bergou *et al.*, 2008) et les *Super-Hélices* (Bertails *et al.*, 2006).

Ce travail a donné lieu à une publication à ACM SIGGRAPH / Transactions on Graphics en 2013 (Casati et Bertails-Descoubes, 2013).

4.1 Modèle de Cosserat, hypothèses de Kirchhoff

Une *tige matérielle* est un solide dont une dimension (longitudinale) est grande devant les deux autres (transverses) définissant la *section de la tige*. Une description mathématique fidèle et compacte de ces solides est donnée par le *modèle de Cosserat* (Cosserat et Cosserat, 1909) que nous présentons maintenant. Nous verrons ensuite que sous certaines hypothèses — d'inextensibilité et de non-cisaillement, ce sont les *hypothèses de Kirchhoff* — simplificatrices mais justifiées, les équations du modèle prennent une forme encore plus simple et compacte. La discrétisation spatiale de ces équations fera l'objet de la prochaine section.

4.1.1 Le modèle de Cosserat

Considérons une tige matérielle ayant une configuration au repos (pas forcément droite) et pouvant subir les déformations élastiques suivantes : flexion, torsion, cisaillement et élongation / compression. Ceci est schématisé figure 4.1.1. La tige est alors décrite

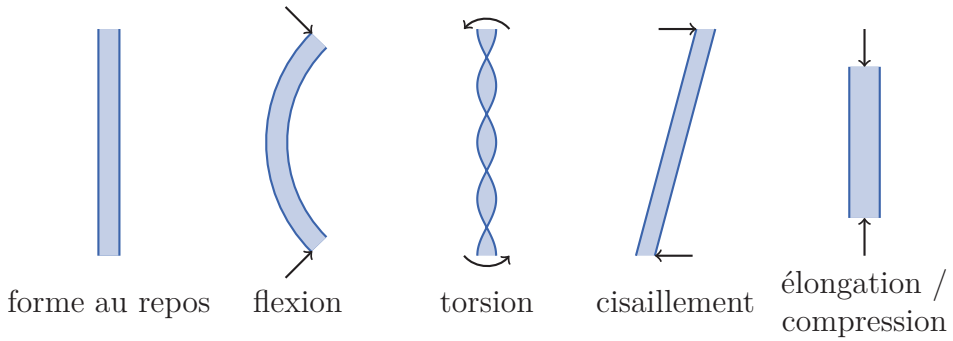


FIGURE 4.1.1 : Les différentes déformations d'une tige de Cosserat.

géométriquement par deux quantités. La *courbe moyenne* $r : [0\,;L] \longrightarrow \mathbb{R}^3$, qui est une paramétrisation du centre de la tige, et la *base matérielle* $\mathcal{R} = (n_0, n_1, n_2) : [0\,;L] \longrightarrow \mathcal{SO}(3)$, une matrice de rotation dont le premier vecteur $n_0(s)$ est normal à la section de la tige en s et les deux autres $(n_1(s), n_2(s))$ forment une base de cette section. Cette base matérielle a pour objet de modéliser la déformation de la section le long de la tige, ce que ne saurait faire la courbe moyenne seule. Le couple $(r, \mathcal{R})$ s'appelle le *repère matériel*, on le note $\mathcal{F}$. Ce formalisme est schématisé figure 4.1.2$_{31}$.

Dans le cas où la base matérielle est de classe $\mathscr{C}^1$, le fait qu'elle reste orthonormée implique l'existence d'un vecteur $\kappa = (\kappa_0, \kappa_1, \kappa_2) : [0\,;L] \longrightarrow \mathbb{R}^3$, continu, permettant de décrire son évolution le long de la tige — voir le théorème D.3.1$_{165}$ pour la démonstration

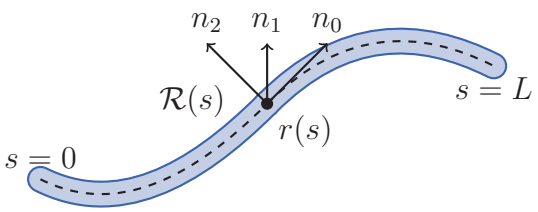


FIGURE 4.1.2 : Une tige matérielle, sa courbe moyenne r (en pointillés) et sa base matérielle $\mathcal{R}$.

de ce résultat. Plus précisément, les variations de la base matérielle sont données par

$$\mathcal{R}'(s) = \mathcal{R}(s)[\kappa(s)]_\times \quad \text{pour } s \in \ [0\,;L] \tag{4.1.3}$$

où le « $'$ » désigne la dérivation par rapport à s et $[u]_\times$ est la matrice (antisymétrique) du produit vectoriel de u à gauche de sorte que

$$[u]_\times v = u \times v \quad \text{pour } u, v \in \mathbb{R}^3. \tag{4.1.4}$$

Si $u = (u_1, u_2, u_3)$, alors

$$[u]_\times = \begin{pmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{pmatrix}. \tag{4.1.5}$$

Le vecteur κ est appelé *vecteur de Darboux à droite* ; le *vecteur de Darboux à gauche*, Ω, étant défini par $\mathcal{R}' = [\Omega]_\times \mathcal{R}$. Ces deux vecteurs sont liés par la relation suivante (*c.f.* théorème D.3.1$_{165}$),

$$\Omega = \mathcal{R}\kappa. \tag{4.1.6}$$

Les vecteurs de Darboux jouent un rôle géométrique important. En effet, ils caractérisent la tendance de la base matérielle à tourner autour des axes de l'espace (dans le cas de Ω) ou de la base elle-même (dans le cas de κ). Par exemple, la première composante de κ, κ_0, quantifie la vitesse à laquelle $\mathcal{R}$ tourne autour de n_0, on l'appelle la *torsion matérielle* ; les deux autres composantes, κ_1 et κ_2, quantifient la vitesse de rotation autour des axes de la section, ce sont les *courbures matérielles* de la tige. Ces quantités ne sont pas à confondre avec la *courbure* et la *torsion géométriques*, données par le repère de Frenet. En effet, ces dernières sont purement géométriques et calculées uniquement à partir de la courbe. Elle ne donnent donc aucune information sur la déformation de la section de la tige comme le fait la base matérielle. L'exemple suivant permet de bien cerner leur différence. Une courbe sans torsion géométrique est nécessairement plane alors qu'une tige avec une torsion matérielle non nulle peut générer une courbe moyenne droite et donc plane (voir la figure 4.1.12$_{33}$). Il est clair que le vecteur de Darboux à droite (κ) joue un rôle plus « intuitif » que son homologue à gauche (Ω). Ceci en fait un candidat privilégié pour la description géométrique de la tige ; dans la suite, on ne parlera presque plus que du vecteur κ et on le nommera le *vecteur de courbures*. Pour mieux comprendre

l'effet de chacune des composantes de κ, on pourra consulter la figure $4.1.12_{33}$ dans le cas restreint d'une *tige de Kirchhoff* (voir plus bas).

En supposant que la courbe moyenne r est elle-aussi de classe $\mathscr{C}^1$, son évolution est décrite par la donnée de sa tangente $t : [0\,;L] \longrightarrow \mathbb{R}^3$ continue et telle que

$$r'(s) = t(s) \quad \text{pour } s \in [0\,;L]. \tag{4.1.7}$$

Finalement, la géométrie d'une tige de Cosserat est complètement modélisée par le *problème de Cauchy* suivant,

$$\begin{cases} \mathcal{F}'(s) = \big(t(s), \mathcal{R}(s)[\kappa(s)]_\times\big) \quad \text{pour } s \in [0\,;L] & \text{(4.1.8a)} \\ \mathcal{F}(0) = \mathcal{F}_0 = (r_0, \mathcal{R}_0) & \text{(4.1.8b)} \end{cases}$$

où l'on voit bien que $\mathcal{F}_0$, t et κ sont les seuls paramètres (indépendants) du problème ; leur connaissance permet à elle seule de connaître toute la configuration de la tige de Cosserat.

Dans la suite, on décrit les *hypothèses de Kirchhoff* qui permettent de s'affranchir de la connaissance de t tout en maintenant la qualité du modèle dans des conditions ayant un large champ d'application.

4.1.2 Les hypothèses de Kirchhoff

À l'échelle macroscopique et pour les systèmes qui nous intéressent, les efforts de flexion/torsion prédominent largement sur les efforts en élongation/compression ou en cisaillement. En effet, par une analyse dimensionnelle, et sous l'hypothèse que le rayon de courbure est comparable à la longueur de la tige, Audoly et Pomeau (2010, section 3.7 p. 90) montrent qu'il n'est pas justifié d'utiliser un modèle considérant tous ces efforts à la fois. Lorsque la flexion/torsion prédomine, les efforts d'élongation/compression et de cisaillement sont négligeables et inversement. Ceci motive et justifie les *hypothèses de Kirchhoff* qui consistent précisément à considérer une tige inextensible et sans cisaillement. L'hypothèse d'inextensibilité se traduit mathématiquement par

$$\|t(s)\| = 1 \quad \text{pour } s \in [0\,;L]. \tag{4.1.9}$$

Ceci implique en particulier que s est une *abscisse curviligne* pour la courbe moyenne r. L'hypothèse de non-cisaillement se traduit par le fait que les sections de la tige restent perpendiculaires à la courbe moyenne, c'est-à-dire,

$$\langle t(s)\,; n_1(s)\rangle = \langle t(s)\,; n_2(s)\rangle = 0 \quad \text{pour } s \in [0\,;L]. \tag{4.1.10}$$

Ceci implique que

$$n_0 = t. \tag{4.1.11}$$

La torsion matérielle τ quantifie désormais la vitesse à laquelle la tige tourne autour d'elle-même. Aussi, on connaît maintenant complètement la géométrie d'une *tige de*

Kirchhoff — une tige de Cosserat sous les hypothèses de Kirchhoff — par la simple connaissance des quantités $\mathcal{F}_0$ et κ.

La figure 4.1.12 présente une tige de Kirchhoff en détaillant l'influence de chacune des composantes du vecteur de courbures κ, donnant ainsi une représentation intuitive de ces quantités.

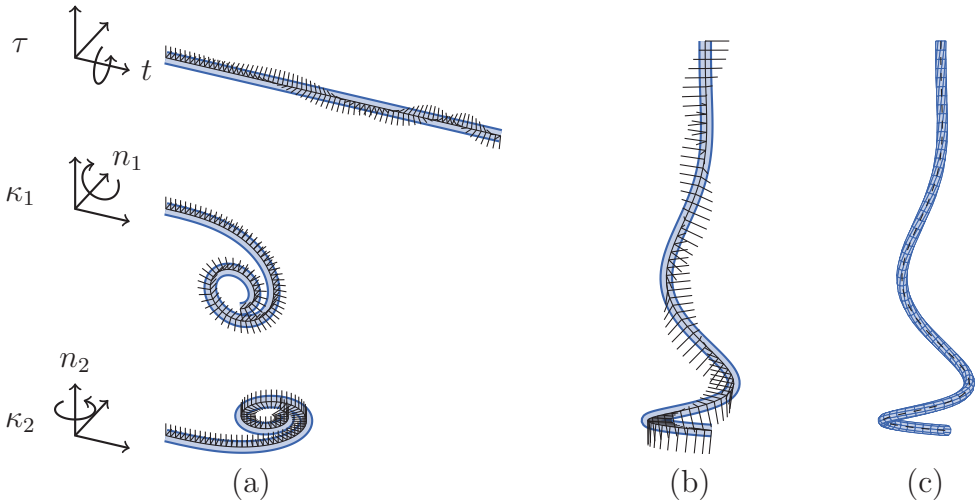


FIGURE 4.1.12 : Variation de la base matérielle le long d'une tige de Kirchhoff. Les composantes du vecteur de courbures varient séparément (a) puis simultanément (b), enfin, la tige tridimensionnelle qui en résulte est représentée (c).

4.1.3 Problème de Darboux

Nous venons de voir qu'une tige de Kirchhoff est décrite complètement par la connaissance des quantités $\mathcal{F}_0$ et κ, où κ est supposé être une fonction continue de s. Ceci nous permet de comprendre que $(\mathcal{F}_0, \kappa)$ forme un jeu minimal de paramètres de la tige, bien adapté à l'élaboration d'un modèle réduit (voir les motivations, chapitre 3_{23}). Le problème suivant, appelé *problème de Darboux* (Darboux, 1887),

$$\text{connaissant } \mathcal{F}_0 \text{ et } \kappa, \text{ trouver } \mathcal{F} \text{ tel que } \begin{cases} \mathcal{F}'(s) = \big(t(s), \mathcal{R}(s)[\kappa(s)]_\times\big) & \forall s \in [0\,;L] \\ \mathcal{F}(0) = \mathcal{F}_0 \end{cases} \tag{4.1.13}$$

possède donc une unique solution, qui nous donne la géométrie de la tige. Cependant, un problème important se pose : on ne connaît pas de solution explicite à ce problème pour un κ quelconque. Voilà donc notre objectif : trouver une forme particulière pour κ de sorte que l'on sache résoudre de manière très précise (voire exacte) le problème de Darboux 4.1.13.

Remarque 4.1.14. La composante rotationnelle du problème de Darboux peut aussi être formulée sur l'espace des quaternions dont on rappelle que la sphère unité est isomorphe

à l'ensemble des rotations de $\mathbb{R}^3$. Sur cet espace, l'équation $\mathcal{R}' = \mathcal{R}[\kappa]_\times$ s'écrit

$$q' = \frac{1}{2}\, q \cdot (0, \kappa) \tag{4.1.15}$$

où $(0, \kappa)$ est le quaternion de partie réelle nulle et de partie imaginaire κ. C'est une manière compacte d'écrire l'équation différentielle du repère mais elle présente un inconvénient : elle rompt la linéarité du problème de Darboux $4.1.13_{33}$. En effet, dans le problème initial $4.1.13_{33}$, l'équation sur le repère est linéaire en κ et celle sur la courbe moyenne est linéaire en $\mathcal{R}$. Sur l'espace des quaternions, l'équation de la courbe moyenne n'est plus linéaire en q puisqu'elle s'écrit

$$r' = q \cdot (0, t) \cdot q^{-1} \tag{4.1.16}$$

où q^{-1} est l'inverse du quaternion q (qui existe toujours puisque $\|q\| = 1$, q représente une rotation). Cette linéarité va être cruciale pour le calcul efficace de la solution du problème de Darboux que nous envisageons. C'est pourquoi, dans la suite nous ne parlerons plus de quaternions et $\mathcal{R}$ sera toujours représentée par une matrice de rotation.

Une expression explicite dans certains cas

La théorie des équations différentielles ordinaires (Demailly, 2006, section 4.1 p. 211) nous dit que dans le cas où $[\kappa(s_1)]_\times$ et $[\kappa(s_2)]_\times$ commutent pour tout $(s_1, s_2) \in [0\,;L]^2$, c'est-à-dire dans le cas où κ est de direction constante, il existe une expression explicite pour le repère matériel, donnée par

$$\mathcal{R}(s) = \mathcal{R}_0 \exp\left(\int_0^s [\kappa(u)]_\times \,\mathrm{d}u\right), \quad \forall s \in [0\,;L]. \tag{4.1.17}$$

La courbe moyenne s'écrit alors

$$r(s) = r_0 + \int_0^s t(u)\,\mathrm{d}u, \quad \forall s \in [0\,;L] \tag{4.1.18}$$

où l'on rappelle que t est le premier vecteur de la base matérielle $\mathcal{R}$. Il est intéressant de constater que ceci implique que le vecteur de Darboux est lui aussi de direction constante, et que son expression devient très simple,

$$\begin{aligned}
\Omega(s) &= \mathcal{R}(s)\kappa(s) && \{\text{équation } 4.1.6_{31}\} && (4.1.19\text{a})\\
&= \mathcal{R}_0 \exp\left(\int_0^s [\kappa(u)]_\times \,\mathrm{d}u\right)\kappa(s) = \mathcal{R}_0 \sum_{n=0}^{+\infty} \frac{1}{n!}\left(\int_0^s [\kappa(u)]_\times \,\mathrm{d}u\right)^n \kappa(s) && && (4.1.19\text{b})\\
&= \mathcal{R}_0\kappa(s) + \mathcal{R}_0 \sum_{n=1}^{+\infty} \frac{1}{n!}\left(\int_0^s [\kappa(u)]_\times \,\mathrm{d}u\right)^{n-1} \int_0^s \underbrace{\kappa(u) \times \kappa(s)}_{0}\,\mathrm{d}u && && (4.1.19\text{c})\\
&= \mathcal{R}_0\kappa(s). && && (4.1.19\text{d})
\end{aligned}$$

On peut interpréter géométriquement cette relation : le repère matériel tourne autour d'une direction constante mais à vitesse variable. Un exemple d'une telle tige se trouve figure $4.1.20_{35}$.

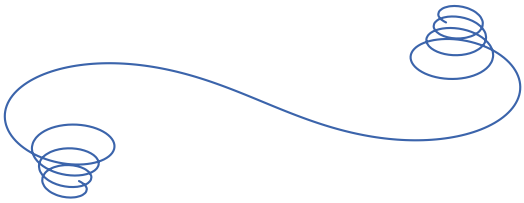


FIGURE 4.1.20 : Une tige de Kirchhoff tridimensionnelle dont le vecteur de courbures est de direction constante mais de norme variable.

Remarque 4.1.21. Notons que dans le cas de la dimension 2, qui correspond à $\tau = \kappa_1 = 0$, κ est bien de direction constante et on peut toujours écrire l'équation $4.1.17_{34}$ mais $4.1.18_{34}$ peut ne pas avoir de solution en forme fermée (voir l'annexe A_{157}) comme c'est le cas dans (Bertails-Descoubes, 2012). De ce point de vue, le problème bidimensionnel est plus simple que le problème en dimension 3.

Remarque 4.1.22. Un autre cas particulier où cette hypothèse est satisfaite est celui des Super-Hélices (Bertails *et al.*, 2006) où le vecteur de courbures est constant. Les équations $4.1.17_{34}$ et $4.1.18_{34}$ possèdent alors une solution en forme fermée (voir l'annexe A_{157}) et les courbes ainsi représentées sont des portions d'hélices circulaires. Elles sont ensuite raccordées de manière $\mathscr{C}^1$ pour former une courbe en hélices par morceaux.

Remarque 4.1.23. Notons en passant qu'il pourrait être intéressant (mais néanmoins limité) de développer le cas de la modélisation d'une tige pour laquelle le vecteur de courbures est de direction constante (ou constante par morceaux). Ceci permettrait d'enrichir le modèle des Super-Hélices. Une telle tige est visible sur la figure 4.1.20.

Malheureusement, dans la plupart des cas, on ne dispose pas de solution explicite et encore moins de forme fermée (voir l'annexe A_{157}) de la géométrie de la tige. Dans la section 4.2_{40} nous donnons une méthode efficace, stable et précise pour le calcul de la géométrie d'une tige de Kirchhoff dans le cas d'une courbure affine par morceaux. Avant cela, nous établissons les équations mécaniques du modèle de Kirchhoff.

4.1.4 Équations mécaniques d'une tige de Kirchhoff

Dans cette section, on établit les équations dynamiques d'une tige de Kirchhoff. Celles-ci sont valides lorsque :

- la longueur de la tige est grande devant les dimensions de sa section,
- la courbure reste modérée (le rayon de courbure est comparable à la longueur de la tige)
- le matériau est homogène (ses propriétés sont les mêmes en tout point) et isotrope (ses propriétés sont les mêmes dans toutes les directions).

4.1.4.1 Principe fondamental de la dynamique appliqué à un élément de tige

Considérons un segment infinitésimal de tige compris entre les sections (droites) d'abscisses s et $s + \delta s$; δs étant amené à tendre vers 0. Voir la figure 4.1.24. Dans un souci

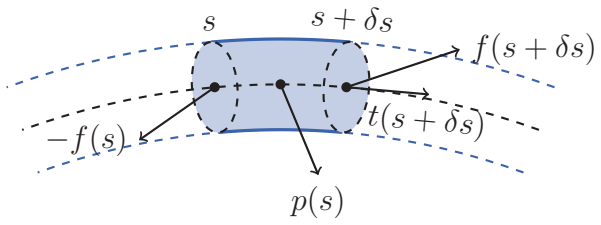


FIGURE 4.1.24 : Bilan des forces appliquées à un segment infinitésimal d'une tige de Kirchhoff comprise entre les sections (droites) s et $s + \delta s$.

de clarté, on ne fait pas apparaître dans les quantités qui suivent leur dépendance en la variable temporelle. Faisons le bilan des forces s'appliquant à la portion de tige considérée :

- la partie en amont ($< s$) exerce sur cette portion une force interne (de tension) que l'on note par convention $-f(s)$; de la même manière, la partie en aval ($> s + \delta s$) exerce une force $f(s + \delta s)$;
- on suppose aussi la présence de forces extérieures (comme le poids ou les contacts) distribuées[1] le long de la tige et modélisées par une densité linéique de force p dont la résultante sur l'élément qui nous intéresse vaut $\int_s^{s+\delta s} p(u)\,\mathrm{d}u = p(s)\delta s + \mathcal{o}(\delta s)$.

Finalement, si l'on note ρ la masse volumique de la tige et S l'aire de sa section (toutes les deux supposées constantes), le principe fondamental de la dynamique en translation appliqué à notre portion de tige donne

$$\rho S \delta s\, \ddot{r}(s) = p(s)\delta s + f(s + \delta s) - f(s) + \mathcal{o}(\delta s). \tag{4.1.25}$$

En divisant par δs et en le faisant tendre vers 0, il vient,

$$\rho S\, \ddot{r} = p + f'. \tag{4.1.26}$$

Faisons maintenant le bilan des moments. On note r_G le centre de gravité de la portion $[s\,;s + \delta s]$.

- La densité de force extérieure, p, applique un moment résultant $\int_s^{s+\delta s} (r(u) - r_G) \times p(u)\,\mathrm{d}u = \mathcal{o}(\delta s)$;

[1] La distribution est ponctuelle dans le cas des forces de contact, ceci est modélisé par une distribution de Dirac.

- la force interne f exerce quant à elle un moment résultant qui s'écrit

$$\begin{aligned}(r(s+\delta s)-r_G)\times f(s+\delta s)+(r(s)-r_G)\times -f(s) &\qquad (4.1.27a)\\ =(r(s)-r_G)\times \delta s f'(s)+\delta s\, t(s)\times f(s+\delta s)+\mathcal{o}(\delta s) &\qquad (4.1.27b)\\ =\delta s\, t(s)\times f(s)+\mathcal{o}(\delta s)\ ; &\qquad (4.1.27c)\end{aligned}$$

- les portions amont et aval transmettent un moment dont la résultante s'écrit $m(s+\delta s)-m(s)$ où l'on a noté $m(s)$ le moment interne transmis par la partie de la tige située entre s et L ;
- enfin, on suppose qu'il n'y a pas de moment extérieur appliqué à la tige.

Finalement, en négligeant la dérivée du moment cinétique de la portion de tige considérée par rapport à la somme des moments appliqués[2], on obtient par le principe fondamental de la dynamique en rotation,

$$m(s+\delta s)-m(s)+\delta s\, t(s)\times f(s)=\mathcal{o}(\delta s) \tag{4.1.28}$$

qui, après division par δs et passage à la limite $\delta s\to 0$, donne

$$m'+t\times f=0. \tag{4.1.29}$$

4.1.4.2 Loi constitutive du matériau

Les équations précédentes ne sont pas suffisantes pour décrire le comportement complet de la tige. En effet, il faut ajouter les équations qui donnent la « réponse » de la tige à une déformation ; ces équations sont appelées *loi de comportement du matériau*. Dans le cas de déformations modérées — les courbures restent comparables à l'inverse de la longueur de la tige L — il est justifié de considérer une loi de comportement *linéaire* :

$$m(s)=\mathcal{R}(s)\,\mathbb{K}\,(\kappa(s)-\bar{\kappa}(s)) \tag{4.1.30}$$

où $\bar{\kappa}$ est le vecteur de courbures de la tige au repos (sans contrainte extérieure) et $\mathbb{K}$ est la matrice de raideur supposée constante et donnée par

$$\mathbb{K}=\begin{pmatrix}\mu J & 0 & 0\\ 0 & EI_1 & 0\\ 0 & 0 & EI_2\end{pmatrix} \tag{4.1.31}$$

où E est le *module de Young*, $\mu=\frac{E}{2(1+\sigma)}$ est le *module de cisaillement*, σ le *coefficient de Poisson*, J le moment d'inertie axial et I_i le moment d'inertie de la section par rapport à l'axe n_i. Par exemple, dans le cas d'une section elliptique de rayon a_i le long de l'axe n_i, ces trois dernières quantités sont données par

$$J=\pi\frac{{a_1}^3{a_2}^3}{{a_1}^2+{a_2}^2},\quad I_1=\frac{\pi}{4}a_1{a_2}^3\quad\text{et}\quad I_2=\frac{\pi}{4}{a_1}^3a_2. \tag{4.1.32}$$

[2]Cette hypothèse découle du caractère négligeable des dimensions de la section devant la longueur de la tige, elle est détaillée dans (Bertails, 2006, section 3.2 p. 126).

4.1.4.3 Synthèse

Rassemblons ici les équations mécaniques d'une tige de Kirchhoff,

$$\begin{cases} \rho S\,\ddot{r} = f' + p & (4.1.33a) \\ m' = f \times t & (4.1.33b) \\ m = \mathcal{R}\,\mathbb{K}(\kappa - \bar{\kappa}) & (4.1.33c) \end{cases}$$

où ρS est la masse linéique de la tige, f est la force interne de la tige, p est la densité linéique de force extérieure, m le moment interne, $\mathbb{K}$ la matrice de raideur et $\bar{\kappa}$ le vecteur de courbures au repos de la tige. Les équations $4.1.13_{33}$ (géométrie) et 4.1.33 (mécanique) constituent les *équations de Kirchhoff*.

Avant de nous interroger sur la résolution de ces équations, nous établissons l'expression de l'énergie interne de la tige qui nous sera utile pour aboutir aux équations de la dynamique dans la section 4.3_{58}.

4.1.4.4 Énergie élastique

On se place dans le cas d'une tige de Kirchhoff dont l'extrémité gauche ($s = 0$) est encastrée (figée en position et en orientation) tandis que l'extrémité droite ($s = L$) est libre (pas de force ni de moment appliqué). Afin d'établir l'expression de l'énergie interne de la tige, E_{int}, calculons d'abord sa variation δE_{int} induite par un déplacement (virtuel) infinitésimal du vecteur de courbures $\delta\kappa$. Pour cela, notons $\delta\mathcal{R}$ et δr les variations induites respectivement sur la base matérielle et sur la courbe moyenne. Le fait que $\mathcal{R}+\delta\mathcal{R}$ reste une matrice de rotation implique[3] l'existence d'un vecteur que nous noterons $\delta\theta$ tel que

$$\delta\mathcal{R} = [\delta\theta]_\times \mathcal{R}. \tag{4.1.34}$$

Toutes les quantités « δ », sauf δE_{int}, dépendent bien entendu de s. Afin de relier les quantités $\delta\kappa$ et $\delta\theta$, on écrit l'équation $(\mathcal{R} + \delta\mathcal{R})' = (\mathcal{R} + \delta\mathcal{R})\,[\kappa + \delta\kappa]_\times$ ce qui donne $[\delta\theta']_\times \mathcal{R} = \mathcal{R}[\delta\kappa]_\times$ soit en utilisant la relation $\mathrm{I}4_{163}$,

$$\delta\theta' = \mathcal{R}\delta\kappa. \tag{4.1.35}$$

Il nous reste enfin à exprimer la variation δr. On écrit $(r + \delta r)' = t + \delta t$ et on obtient

$$\delta r' = \delta\theta \times t. \tag{4.1.36}$$

[3]Pour voir ceci, on montre que $\delta\mathcal{R}\mathcal{R}^\top + \mathcal{R}\delta\mathcal{R}^\top = 0$ et donc que $\delta\mathcal{R}\mathcal{R}^\top$ est antisymétrique, voir le théorème $\mathrm{D.3.1}_{165}$.

Nous avons maintenant tous les ingrédients pour exprimer la variation δE_{int} de l'énergie interne. C'est l'opposé du travail de la force interne (linéique) qui s'écrit

$$\delta E_{\text{int}} = -\int_0^L \langle f' \,;\, \delta r \rangle \,\mathrm{d}s = \int_0^L \langle f \,;\, \delta r' \rangle \,\mathrm{d}s - \left[\langle f \,;\, \delta r \rangle\right]_0^L \tag{4.1.37a}$$

$$\overset{4}{=} \int_0^L \langle f \,;\, \delta\theta \times t \rangle \,\mathrm{d}s \overset{5}{=} -\int_0^L \langle f \times t \,;\, \delta\theta \rangle \,\mathrm{d}s \tag{4.1.37b}$$

$$= -\int_0^L \langle m' \,;\, \delta\theta \rangle \,\mathrm{d}s = \int_0^L \langle m \,;\, \delta\theta' \rangle \,\mathrm{d}s - \left[\langle m \,;\, \delta\theta \rangle\right]_0^L \overset{6}{=} \int_0^L \langle m \,;\, \mathcal{R}\delta\kappa \rangle \,\mathrm{d}s \tag{4.1.37c}$$

$$= \int_0^L \langle \mathbb{K}(\kappa - \bar{\kappa}) \,;\, \delta\kappa \rangle \,\mathrm{d}s. \tag{4.1.37d}$$

Ceci nous permet alors de donner l'expression de l'énergie interne

$$E_{\text{int}}(\kappa, \bar{\kappa}) = \frac{1}{2} \int_0^L \|\kappa - \bar{\kappa}\|_{\mathbb{K}}^2 \,\mathrm{d}s \tag{4.1.38}$$

puisque par convention l'énergie interne est nulle au repos ($\kappa = \bar{\kappa}$).

4.1.5 Résolution des équations de Kirchhoff

Nous avons déjà vu que le problème géométrique $4.1.13_{33}$ n'admettait pas de solution explicite dans le cas général ; c'est donc aussi le cas des équations de Kirchhoff $4.1.33_{38}$. Il est ainsi nécessaire de considérer des méthodes d'approximation des solutions.

Remarquons que le problème $4.1.33_{38}$ n'est pas un problème de Cauchy — ou de manière plus explicite mais moins française, un *initial value problem*, c'est-à-dire un problème où toutes les conditions initiales sont connues — comme l'était le problème géométrique $4.1.13_{33}$. En effet, les inconnues du problème $4.1.33_{38}$ sont f, m et κ — r et t se déduisant de κ par résolution du problème géométrique $4.1.13_{33}$, au moins lorsque $\mathcal{F}_0$ est connu — et ni $f(0)$, ni $m(0)$, ni $\kappa(0)$ ne sont connus.

En revanche, dans le cas où l'extrémité $s = L$ de la tige est « libre », le moment et la force extérieure y sont nécessairement nuls — la partie avale de la tige ne transmet ni de force ni de moment puisqu'elle est vide. Notons que ceci implique qu'à cette extrémité, $\kappa(L) = \bar{\kappa}(L)$ d'après $4.1.33c_{38}$. Si l'autre extrémité est en plus *encastrée*, une partie de l'information est donc connue en $s = 0 : \mathcal{F}(0) = \mathcal{F}_0$, ainsi qu'en $s = L : m(L) = f(L) = 0$, $\kappa(L) = \bar{\kappa}(L)$. C'est un problème aux limites — ou de manière tout aussi explicite mais plus anglaise, un *boundary value problem* — non linéaire. Ces problèmes sont réputés pour être difficiles. Ceci explique que ce n'est seulement que récemment que l'on a

[4] Le crochet de l'intégration par parties est nul puisque $f(L) = 0$ (l'extrémité droite est libre) et $\delta r(0) = 0$ (l'extrémité gauche est encastrée).

[5] On a utilisé l'identité $I5_{163}$.

[6] Le crochet dans l'expression de gauche est nul puisque $m(L) = 0$ (l'extrémité droite est libre) et $\delta\theta(0) = 0$ (l'extrémité gauche est encastrée ce qui implique $0 = \delta\mathcal{R}_0 = \mathcal{R}(0)[\delta\theta(0)]_\times$).

développé des méthodes efficaces et robustes pour leur résolution dans le cas dynamique (Bertails *et al.*, 2006; Bergou *et al.*, 2008).

Dans les sections suivantes, nous détaillons les aspects géométriques puis dynamiques de notre approche, avant de la comparer aux modèles de (Bertails *et al.*, 2006) et (Bergou *et al.*, 2008) dans le chapitre suivant.

4.2 Géométrie d'une Clothoïde 3D

Nous nous consacrons dans cette section à la résolution numérique approchée du problème de Darboux $4.1.13_{33}$. L'approche est la suivante. En regardant le problème de Darboux sur un sous-espace du vecteur des courbures, appelé espace des *primitives*, on peut espérer aboutir à un problème plus simple et trouver une solution dans certains cas. Si ce sous-espace est dense par « raccordement » des primitives, on aura alors trouvé une méthode d'approximation des solutions du problème de Darboux.

Le cas qui nous intéressera finalement sera celui où les primitives sont des fonctions affines de l'abscisse curviligne, la courbe résultante ayant un vecteur de courbures continu et affine par morceaux. Cependant, nous commençons par énoncer quelques résultats pour une classe de fonctions plus générales, celles des fonctions développables en série entière en 0.

4.2.1 Solution développable en série entière

Il est intéressant de constater que dès lors que le vecteur de courbures est développable en série entière, la solution du problème de Darboux l'est aussi ; de plus, la convergence de la série solution est très rapide. Ceci fait l'objet du théorème ci-dessous. Nous ne pouvons que recommander au lecteur la lecture de sa démonstration, car les majorations qui y sont effectuées sont au cœur de notre algorithme de calcul efficace par séries entières.

Théorème 4.2.1. *Soit $\kappa : \mathbb{R} \longrightarrow \mathbb{R}^3$ développable en série entière autour de 0 sur $\mathbb{R}$ tout entier (on dit que le rayon de convergence de la série est infini). Ceci signifie que κ satisfait*

$$\kappa(s) = \sum_{n=0}^{+\infty} \lambda_n s^n \quad \forall s \in \mathbb{R}, \tag{4.2.2}$$

où les λ_n[7] sont des vecteurs de $\mathbb{R}^3$. Alors l'unique solution du problème de Darboux $4.1.13_{33}$ est elle aussi développable en série entière sur $\mathbb{R}$,

$$\mathcal{R}(s) = \sum_{n=0}^{+\infty} \mathcal{R}_n s^n \quad \text{et} \quad r(s) = \sum_{n=0}^{+\infty} r_n s^n, \qquad \forall s \in \mathbb{R}, \tag{4.2.3}$$

[7] L'utilisation des notations κ_1 et κ_2 pour les courbures matérielles (section $4.1.1_{30}$) nous empêche de désigner par κ_n les coefficients de la série κ, nous choisissons de les noter λ_n.

et les coefficients satisfont les équations de récurrence suivantes, $\forall n \in \mathbb{N}$,

$$\begin{cases} \mathcal{R}_{n+1} = \dfrac{1}{n+1} \displaystyle\sum_{k=0}^{n} \mathcal{R}_k [\lambda_{n-k}]_\times & (4.2.4a) \\ r_{n+1} = \dfrac{1}{n+1} t_n, & (4.2.4b) \end{cases}$$

où t_n *est la première colonne de* $\mathcal{R}_n$*. De plus, la convergence de ces séries est rapide puisque les restes sont majorés par des suites géométriques dont la raison peut être choisie arbitrairement proche de* 0*. Plus précisément, pour la base matérielle par exemple, si l'on fixe* $s \in \mathbb{R}$*, que l'on pose* $C(s) = 2|s| \sum_{n=0}^{+\infty} \|\lambda_n s^n\|$*, et* $M_s = \frac{C(s)^{\lfloor C(s) \rfloor}}{\lfloor C(s) \rfloor !}$*, alors* $\forall \omega \in]0\,;1[$ *et* $\forall n \in \mathbb{N}$,[8]

$$\left\| \mathcal{R}(s) - \sum_{k=0}^{n-1} \mathcal{R}_k s^k \right\| \leqslant M_{\frac{s}{\omega}} \frac{\omega^n}{1-\omega} \leqslant e^{C\left(\frac{s}{\omega}\right)} \frac{\omega^n}{1-\omega} \tag{4.2.5}$$

où $\|\cdot\|$ *désigne la norme du maximum des valeurs absolues,* $\|A\| = \sup_{i,j} |a_{ij}|$, $A \in \mathcal{M}_3(\mathbb{R})$.

---

dem. Nous avons retrouvé ce résultat mais il est en grande partie montré dans (Neher, 1996) pour le cas scalaire et la démonstration présentée ici est proche de celle de Neher (1999). La majoration 4.2.5 sous cette forme précise est de notre cru (à notre connaissance) et sera cruciale dans la suite. On pourra consulter le dernier paragraphe de la section 4.2.2.2$_{50}$ pour une discussion plus détaillée de nos contributions propres sur ce point. Pour mieux comprendre cette démonstration, il peut être intéressant d'avoir en tête le « profil de bosse » typique des coefficients de la série, que nous étudions dans la section 4.2.2$_{45}$ et qui sont visibles figure 4.2.29$_{49}$.

Tout d'abord, il est clair que si la base matérielle est développable en série entière, la courbe moyenne r l'est aussi et son développement satisfait la récurrence précédente. Il suffit donc de prouver que la série définie par la récurrence 4.2.4a est convergente et que sa limite est solution de l'équation $\mathcal{R}' = \mathcal{R}[\kappa]_\times$.

Montrons que la série définie par 4.2.4a converge. Il suffit pour cela de montrer qu'elle converge absolument (par complétude de $(\mathcal{M}_3(\mathbb{R}), \|\cdot\|)$). Notons que si $A, B \in \mathcal{M}_3(\mathbb{R})$ avec B antisymétrique, alors $\|AB\| \leqslant 2\|A\|\|B\|$. On note $\Gamma(n,s) = \max_{k \leqslant n} \left\{ \left\| \mathcal{R}_k s^k \right\| \right\}$, $\forall n \in \mathbb{N}$ et $\forall s \in \mathbb{R}$; d'après la récurrence sur $\mathcal{R}_n$,

$$\left\| \mathcal{R}_{n+1} s^{n+1} \right\| \leqslant \frac{2|s|}{n+1} \sum_{k=0}^{n} \left\| \mathcal{R}_k s^k \right\| \left\| \lambda_{n-k} s^{n-k} \right\| \leqslant \frac{2|s|}{n+1} \Gamma(n,s) \sum_{k=0}^{+\infty} \left\| \lambda_k s^k \right\| \leqslant \frac{C(s)}{n+1} \Gamma(n,s) \tag{4.2.6}$$

où l'on a posé $C(s) = 2|s| \sum_{n=0}^{+\infty} \|\lambda_n s^n\|$ qui est finie puisque $\sum \lambda_n s^n$ est convergente donc absolument convergente (c'est le lemme d'Abel, il est rappelé en annexe B$_{159}$). Posons

[8] Par convention, une somme qui ne contient aucun terme est nulle. C'est le cas ici lorsque $n = 0$. On obtient alors un majorant de la somme absolue.

$n_0(s) = \lfloor C(s) \rfloor$ de sorte que $\forall n \geqslant n_0(s)$, $\frac{C(s)}{n+1} < 1$; ceci implique $\Gamma(n,s) = \Gamma(n_0(s),s)$ d'après la majoration précédente ; on a donc $\|\mathcal{R}_n s^n\| \leqslant \Gamma(n_0(s),s)$, $\forall n \in \mathbb{N}$. Le terme général de la série est donc borné, ceci prouve que la série $\sum \|\mathcal{R}_n u^n\|$ converge pour $|u| < |s|$ (c'est encore une fois le lemme d'Abel). Ceci étant vrai pour tout $s \in \mathbb{R}$ et l'espace étant complet, la série $\sum \mathcal{R}_n s^n$ converge bien sur $\mathbb{R}$ (pour un rappel de ce résultat, on pourra aussi consulter l'annexe B_{159}).

Majorons maintenant le reste de la série. Pour cela, notons $\Gamma_s = \Gamma(n_0(s),s)$ et fixons $\omega \in \,]0\,;1[$, $s \in \mathbb{R}$ et $n \in \mathbb{N}$,

$$\left\| \mathcal{R}(s) - \sum_{k=0}^{n-1} \mathcal{R}_k s^k \right\| \leqslant \sum_{k=n}^{+\infty} \left\| \mathcal{R}_k s^k \right\| \leqslant \sum_{k=n}^{+\infty} \left\| \mathcal{R}_k \left(\frac{s}{\omega}\right)^k \right\| \omega^k \leqslant \sum_{k=n}^{+\infty} \Gamma_{\frac{s}{\omega}} \omega^k \leqslant \Gamma_{\frac{s}{\omega}} \frac{\omega^n}{1-\omega}. \tag{4.2.7}$$

Pour aboutir aux inégalités $4.2.5_{41}$, il nous reste à montrer les majorations $\Gamma_s \leqslant M_s = \frac{C(s)^{n_0(s)}}{n_0(s)!} \leqslant e^{C(s)}$. Prenons $n < n_0(s)$, on a $\frac{C(s)}{n+1} \geqslant 1$ et il vient,

$$\Gamma(n+1,s) = \max\left(\Gamma(n,s), \left\|\mathcal{R}_{n+1} s^{n+1}\right\|\right) \leqslant \Gamma(n,s) \max\left(1, \frac{C(s)}{n+1}\right) \leqslant \Gamma(n,s) \frac{C(s)}{n+1}, \tag{4.2.8}$$

d'où, par une récurrence finie et en observant que $\Gamma(0,s) = \|\mathcal{R}_0\| \leqslant 1$ ($\mathcal{R}_0$ est une matrice de rotation), on obtient $\Gamma_s = \Gamma(n_0(s),s) \leqslant \frac{C(s)^{n_0(s)}}{n_0(s)!}$. Ceci donne la première inégalité de $4.2.5_{41}$. Pour obtenir la seconde inégalité, il suffit de remarquer que $\frac{C(s)^{n_0(s)}}{n_0(s)!}$ est un terme de la série $e^{C(s)}$, il est donc majoré par la somme (c'est une série à termes positifs puisque $C(s) \geqslant 0$). En utilisant l'égalité $4.2.4\mathrm{b}_{41}$, on majore de la même manière les restes de la série $\sum r_n s^n$.

Il reste à montrer que si la base matérielle est développable en série entière alors elle satisfait la récurrence $4.2.4\mathrm{a}_{41}$. Pour cela, « injectons » l'expression $\mathcal{R}(s) = \sum_{n=0}^{+\infty} \mathcal{R}_n s^n$ dans l'équation différentielle $\mathcal{R}' = \mathcal{R}[\kappa]_\times$,

$$\sum_{n=1}^{+\infty} n \mathcal{R}_n s^{n-1} = \sum_{n=0}^{+\infty} \sum_{k=0}^{n} \mathcal{R}_k [\lambda_{n-k}]_\times s^n \implies \sum_{n=0}^{+\infty} \left((n+1)\mathcal{R}_{n+1} - \sum_{k=0}^{n} \mathcal{R}_k [\lambda_{n-k}]_\times \right) s^n = 0, \tag{4.2.9}$$

ce qui, par unicité du développement en série entière de la fonction nulle, donne la récurrence $4.2.4\mathrm{a}_{41}$. □

4.2.1.1 Sommation finie

La convergence rapide vers 0 des restes de la série est un réel atout pour espérer aboutir à une méthode efficace de la résolution du problème de Darboux $4.1.13_{33}$. La majoration $4.2.5_{41}$ permet même de connaître à l'avance le nombre de termes nécessaire pour garantir une précision donnée (si l'on néglige les erreurs d'arrondi et autres problèmes numériques

comme ceux que nous rencontrons plus loin). En effet, si l'on note ε la précision que l'on cherche à atteindre, alors, $\forall s \in \mathbb{R}$ et $\forall \omega \in \,]0\,;1[$, le reste est majoré par ε dès lors que $e^{C\left(\frac{s}{\omega}\right)} \frac{\omega^n}{1-\omega} \leqslant \varepsilon$, c'est-à-dire, lorsque

$$n \geqslant \frac{-C\left(\frac{s}{\omega}\right) + \ln(1-\omega) + \ln \varepsilon}{\ln \omega} = f(\omega). \tag{4.2.10}$$

De cette dernière majoration on peut tirer une loi intuitive : le nombre minimum de termes à sommer pour obtenir une précision fixée est une fonction (au plus) linéaire de $C\left(\frac{s}{\omega}\right)$. On peut se demander comment varie cette quantité en fonction de ω : prendre une valeur proche de zéro pour ω fait « exploser » la quantité $C\left(\frac{s}{\omega}\right)$; une valeur proche de 1 a le même impact sur $\frac{1}{\ln \omega}$. Le profil de la fonction $\omega \longmapsto f(\omega)$ dans le cas d'une courbure polynomiale (de degré 1) est donné figure 4.2.11. Il existe en fait une valeur optimale

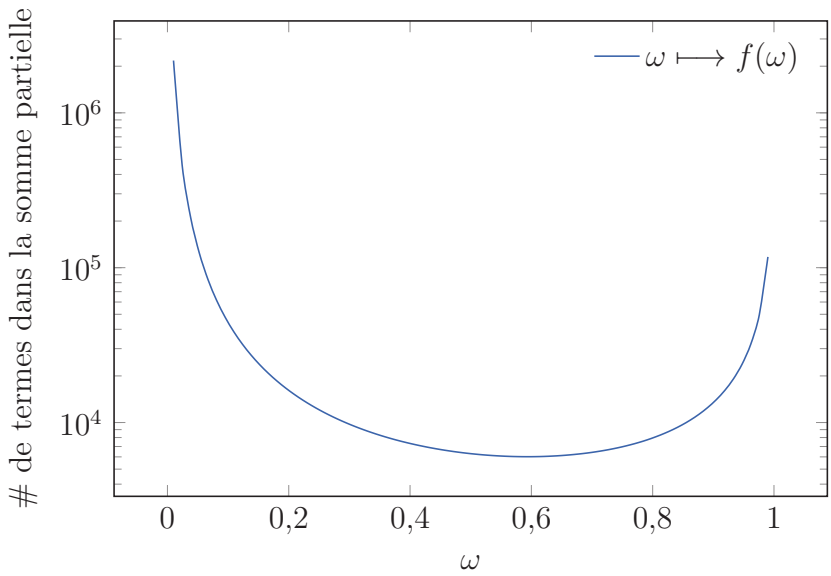


FIGURE 4.2.11 : Nombre de termes qu'il suffit de considérer dans $4.2.5_{41}$, d'après la majoration $4.2.5_{41}$, pour garantir une précision ε en fonction du paramètre ω. Ici, $C(s) = s(10s + 20)$, $s = 10$ et $\varepsilon = 2{,}22 \cdot 10^{-16}$.

du paramètre ω pour laquelle la majoration $4.2.5_{41}$ est la plus fine[9]. On peut la calculer numériquement en minimisant la fonction f (ou par dichotomie sur f') mais prenons plutôt un exemple et fixons $\omega = \frac{1}{2}$. Supposons que $\varepsilon = 2^{-m}$ est la précision machine ; m étant la taille de la mantisse, $m = 23$ en simple précision, $m = 52$ en double. La loi précédente s'écrit alors très simplement

$$n \geqslant \frac{1}{\ln 2} C(2s) + m + 1. \tag{4.2.12}$$

[9]Pour être encore plus précis, il faudrait travailler avec la quantité M_s plutôt qu'avec son majorant (un peu fort) $e^{C(s)}$. L'utilisation de ce dernier a l'avantage pédagogique de faire apparaître clairement la linéarité en $C\left(\frac{s}{\omega}\right)$ dans 4.2.10.

Dans le cas où κ est un polynôme de degré 1, que nous détaillons dans la section $4.2.2_{45}$, une centaine de termes sont nécessaires pour une approximation raisonnable de la somme de la série. Avec l'algorithme de découpes que nous donnons section $4.2.2.2_{50}$, seulement une vingtaine de termes seront nécessaires. Même si ce découpage est envisagé pour éviter les problèmes de précision flottante, il a aussi l'avantage de réduire le nombre de termes nécessaires à l'évaluation de la somme sur chaque segment.

Finalement, on a montré que le fait d'utiliser des sommations infinies n'est pas rédhibitoire pour une implémentation des récurrences $4.2.4_{41}$. En effet, celles-ci ne pourront être itérées qu'un petit nombre de fois tout en maîtrisant l'erreur commise, du moins en arithmétique exacte.

4.2.1.2 Cas des courbures polynomiales

Lorsque κ est polynomial, le théorème $4.2.1_{40}$ affirme que la géométrie de la tige matérielle est développable en série entière sur $\mathbb{R}$ tout entier et la récurrence $4.2.4\mathrm{a}_{41}$ est finie. Elle est même linéaire à $d+1$ termes si le polynôme κ est de degré d. Ceci semble constituer un algorithme simple et efficace, notamment grâce à la sommation finie du paragraphe précédent, pour le calcul de la géométrie d'une tige dont les courbures sont polynomiales. Malheureusement, de tels calculs posent de gros problèmes numériques même dans le cas $d = 1$ d'une fonction affine. Ce dernier cas fait l'objet de la section suivante pour lequel nous donnons une méthode efficace, précise et stable pour évaluer la somme de la série solution du problème de Darboux.

Dans la section $4.2.4_{55}$, on montre que notre algorithme se généralise bien au degré $d > 1$ mais qu'il impose néanmoins quelques difficultés d'implémentation, pour la simulation dynamique notamment.

Remarque 4.2.13. Lorsque le vecteur de courbures est polynomial, on peut se demander s'il en est de même pour le vecteur de Darboux à gauche, Ω, de l'équation $4.1.6_{31}$. Ce qui est toujours vrai, c'est que $\Omega' = \mathcal{R}'\kappa + \mathcal{R}\kappa' = \mathcal{R}\kappa \times \kappa + \mathcal{R}\kappa'$, et donc

$$\Omega' = \mathcal{R}\kappa'. \tag{4.2.14}$$

On retrouve[10] ici que dans le cas d'un vecteur de courbures indépendant de s, le vecteur Ω est lui aussi constant (Bertails *et al.*, 2006). En revanche, dans le cas plus général d'un vecteur de courbures polynomial, Ω n'est pas nécessairement polynomial comme en atteste la figure $4.2.15_{45}$ dans le cas affine. Notons aussi que dans le cas particulier d'un vecteur de courbures de direction constante (c'est toujours le cas en dimension 2) et polynomial, le vecteur Ω est lui aussi de direction constante et polynomial, puisqu'on montre (par récurrence) l'égalité suivante[10] sur les dérivées n-ièmes,

$$\Omega^{(n)} = \mathcal{R}\kappa^{(n)}, \qquad \forall n \in \mathbb{N}. \tag{4.2.16}$$

[10] Une autre démonstration est donnée par l'équation $4.1.19\mathrm{d}_{34}$.

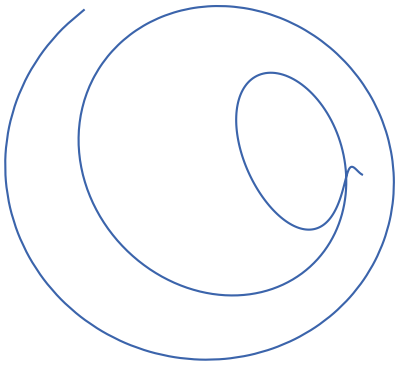


FIGURE 4.2.15 : Le fait que le vecteur de courbures soit une fonction polynomiale n'implique pas que le vecteur de Darboux à gauche ($4.1.6_{31}$) le soit aussi. Ici, on a tracé l'extrémité du vecteur de Darboux à gauche (Ω) pour la tige de la figure $4.1.12_{33}$ dont le vecteur de courbures est une fonction affine.

4.2.2 Séries entières pour la Clothoïde 3D, compensation catastrophique, remède

Le cas qui nous intéresse ici est celui où κ est un polynôme de degré 1, une fonction affine s'écrivant

$$\kappa(s) = \lambda_0 + s\lambda_1, \qquad \forall s \in [0\,;L]. \tag{4.2.17}$$

Dans ce cas, la géométrie de la tige est développable en série entière et le système $4.2.4_{41}$ donne les relations de récurrence suivantes,

$$\begin{cases} \mathcal{R}_1 = \mathcal{R}_0[\lambda_0]_\times & \text{(4.2.18a)} \\ \mathcal{R}_{n+2} = \dfrac{1}{n+2}\left(\mathcal{R}_{n+1}[\lambda_0]_\times + \mathcal{R}_n[\lambda_1]_\times\right) & \text{(4.2.18b)} \\ r_{n+1} = \dfrac{1}{n+1}t_n. & \text{(4.2.18c)} \end{cases}$$

Dans le reste de cette section, nous montrons dans un premier temps qu'une implémentation naïve de ces récurrences mène à de sévères problèmes numériques, puis, nous présentons notre algorithme de calcul permettant de remédier à ces problèmes et de garantir une précision numérique élevée.

4.2.2.1 Calcul numérique des séries, compensation catastrophique

La difficulté du calcul de la géométrie par série entière réside surtout dans l'application de la récurrence 4.2.18b, la récurrence 4.2.18c étant numériquement stable. Dans la suite, nous nous consacrerons donc essentiellement au calcul de la récurrence sur les termes de la base, 4.2.18b.

Plutôt que de calculer séparément les termes de la récurrence 4.2.18b$_{45}$ et les puissances s^n et d'en faire le produit, on préfère[11] calculer directement les termes $\mathcal{R}_n s^n$. Pour cela, on transforme la récurrence sur $\mathcal{R}_n$ en une récurrence sur $\mathcal{R}_n s^n$ que nous noterons désormais $\mathcal{R}_n(s)$. Lorsque la dépendance en s de $\mathcal{R}_n$ est explicite comme dans $\mathcal{R}_n(s)$, cette expression désigne en fait $\mathcal{R}_n s^n$; lorsqu'elle n'est pas présente, on fait bien référence à $\mathcal{R}_n$. Cette notation ne devrait pas ajouter d'ambiguïté. La récurrence 4.2.18b$_{45}$ devient

$$\begin{cases} \mathcal{R}_1(s) = s\mathcal{R}_0[\lambda_0]_\times & \text{(4.2.19a)} \\ \mathcal{R}_{n+2}(s) = \dfrac{s}{n+2}\left(\mathcal{R}_{n+1}(s)[\lambda_0]_\times + s\mathcal{R}_n(s)[\lambda_1]_\times\right). & \text{(4.2.19b)} \end{cases}$$

Le théorème 4.2.1$_{40}$ affirme que la convergence de la série $\sum \mathcal{R}_n(s)$ est rapide. Rappelons-nous que la majoration des restes de la série entière par des suites géométriques qui convergent rapidement vers 0 implique qu'il n'est pas nécessaire de sommer beaucoup de termes pour obtenir une bonne estimation de la somme. Ceci est de bonne augure pour une implémentation efficace du calcul de la géométrie.

Cependant, en pratique, l'implémentation naïve de la récurrence 4.2.19 donne lieu à de grosses instabilités numériques comme sur la figure 4.2.20(a). Plus précisément, c'est

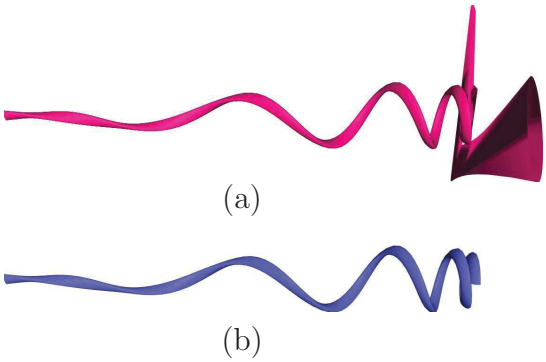


FIGURE 4.2.20 : (a) Problème de compensation catastrophique lors de l'utilisation de la récurrence 4.2.19. Les paramètres géométriques sont pourtant relativement modérés : $L = \frac{1}{2}$, $\|\kappa(s)\| \in [0\,;100]$. (b) Précision numérique retrouvée en utilisant notre algorithme.

un phénomène de *compensation catastrophique* qui en est à l'origine. Nous en donnons la définition dans le paragraphe suivant. Puis, nous mettons en évidence ce phénomène sur un cas simple (mais proche de celui de la récurrence 4.2.19) avant d'y remédier dans la section qui suit.

La *compensation catastrophique* Le phénomène de compensation catastrophique (Goldberg, 1991) apparaît lorsqu'en arithmétique flottante, on soustrait deux valeurs très

[11] En effet, pour $s = 100$ par exemple, le terme de rang 155, s^{155} dépasse déjà la plus grande valeur que peut prendre un flottant en double précision ($\approx 2{,}23 \cdot 10^{308}$).

proches comportant des erreurs d'arrondi. Par exemple, dans le calcul de $(x+y)-x$, la somme $x+y$ peut être soumise à des erreurs d'arrondi si y est proche de 0 relativement à $|x|$. Un cas extrême mais pédagogique est le suivant, $x = \frac{1}{\varepsilon}$ et $y = 1$, ε désignant la précision machine (de l'ordre de $1{,}2 \cdot 10^{-7}$ en simple précision, $2{,}2 \cdot 10^{-16}$ en double). Dans ce cas, $x+y = \frac{1}{\varepsilon}+1 \approx \frac{1}{\varepsilon} \approx x$, d'où $(x+y)-x \approx 0 \neq y$, l'erreur relative est donc de 100% ! Retenons que la soustraction flottante de deux valeurs proches et soumises aux erreurs d'arrondi, comme c'est le cas pour des quantités issues d'un calcul numérique, est dangereuse numériquement.

Mise en évidence du phénomène de compensation catastrophique dans nos calculs Dans ce paragraphe, nous considérons un exemple simple mais présentant les mêmes difficultés numériques que celles posées par la récurrence $4.2.19_{46}$. Le but étant que le lecteur identifie l'origine de l'apparition de la compensation catastrophique dans notre problème.

On considère l'équation différentielle sur $\mathbb{R}$ suivante,

$$Y' = YJ \quad \text{avec} \quad J = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}. \tag{4.2.21}$$

Sa solution est connue explicitement, c'est la matrice de rotation

$$Y(s) = \exp(sJ) = \begin{pmatrix} \cos s & -\sin s \\ \sin s & \cos s \end{pmatrix} \qquad \forall s \in \mathbb{R}, \tag{4.2.22}$$

pour la condition initiale $Y(0) = I_2$. Cependant, si l'on tente de résoudre cette équation différentielle par la méthode des séries entières, on aboutit à la récurrence (du même type que $4.2.19_{46}$) suivante

$$Y_{n+1}(s) = \frac{s}{n+1} Y_n(s) J. \tag{4.2.23}$$

Une implémentation naïve de cette récurrence en Python/NumPy est donnée dans l'algorithme $4.2.24_{48}$.

Cette implémentation nous permet de tracer l'erreur numérique en fonction de s, que l'on note $\|Y(s) - \sum Y_n(s)\|$, puisque la solution exacte Y est connue exactement (4.2.22). Ce tracé est effectué figure $4.2.25_{48}$, il met en évidence une « explosion » de l'erreur avec s. Essayons de comprendre pourquoi.

La norme de $Y_n(s)$ suit la même récurrence que les termes de e^s, en effet,

$$\|Y_{n+1}(s)\| = \frac{s}{n+1} \|Y_n(s)\|. \tag{4.2.26}$$

C'est-à-dire que la norme des coefficients est une suite pseudo-géométrique, *i.e.* dont la (pseudo-)raison dépend de n. La suite des normes est donc croissante lorsque cette raison est plus grande que 1, décroissante sinon. Dans notre cas, la pseudo-raison vaut $\frac{s}{n+1}$ et décroît vers 0 ce qui garantit que $\|Y_n(s)\|$ va décroître à partir d'un certain rang ; c'est

Algorithme 4.2.24 – Implémentation en Python/NumPy de la résolution de $4.2.21_{47}$ par la récurrence $4.2.23_{47}$.

```
from __future__ import division
import numpy as np

j, yn = np.array([[0, -1], [1, 0]]), np.eye(2)
s, y = 50, yn
nmax = np.ceil(2 * s / np.log(2) + 53)  # c.f. equation 4.2.12_43

for n in range(int(nmax)):
    yn = np.dot(yn, j) * (s / (n + 1))
    y += yn
```

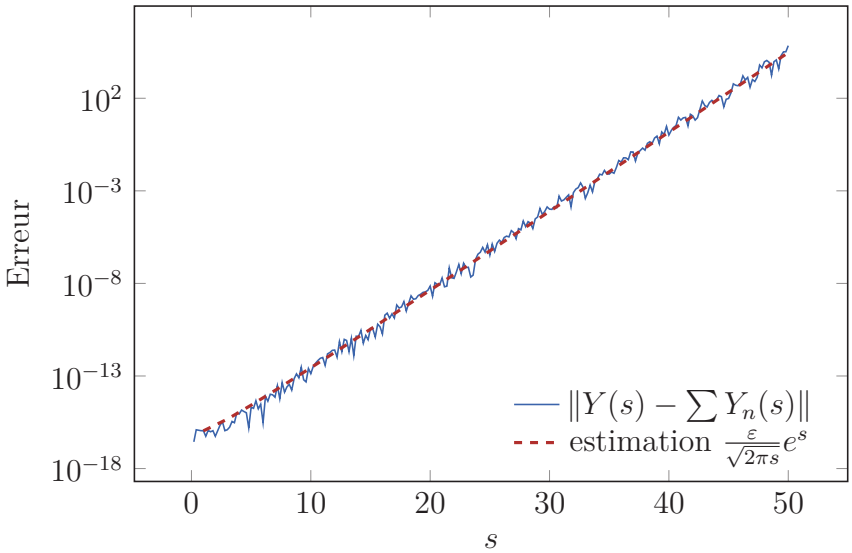


FIGURE 4.2.25 : Erreur numérique commise par l'algorithme 4.2.24 pour la résolution de l'équation $4.2.21_{47}$. L'équation $4.2.28_{49}$ donne une bonne estimation de cette l'erreur (ε est la précision flottante).

ce que nous avions déjà mis en évidence dans la preuve du théorème $4.2.1_{40}$. Cependant, avant d'atteindre ce rang, les termes sont croissants ; c'est un point clef pour comprendre la compensation catastrophique qui a lieu ici. Si l'on note $n_0 = \lfloor s \rfloor$ la partie entière par défaut de s, on montre[12] que $Y_{n_0}(s)$ majore tous les $Y_n(s)$,

$$\|Y_n(s)\| \leqslant \|Y_{n_0}(s)\| \qquad \forall n \in \mathbb{N}. \tag{4.2.27}$$

[12] Ceci est en fait montré dans la preuve du théorème $4.2.1_{40}$.

De plus, on a une estimation asymptotique en fonction de s de ce majorant,

$$\|Y_{n_0}(s)\| = \left\|Y_{\lfloor s \rfloor}(s)\right\| = \frac{s^{\lfloor s \rfloor}}{\lfloor s \rfloor !} = \Theta\left(\frac{e^s}{\sqrt{2\pi s}}\right) \tag{4.2.28}$$

où la notation Θ est rappelée lors de l'introduction des notations page 15. La norme des termes de la série atteint donc un maximum au rang n_0 et la valeur de ce maximum croît exponentiellement avec s. La figure 4.2.29 met en évidence cette croissance exponentielle du majorant des coefficients par rapport à s.

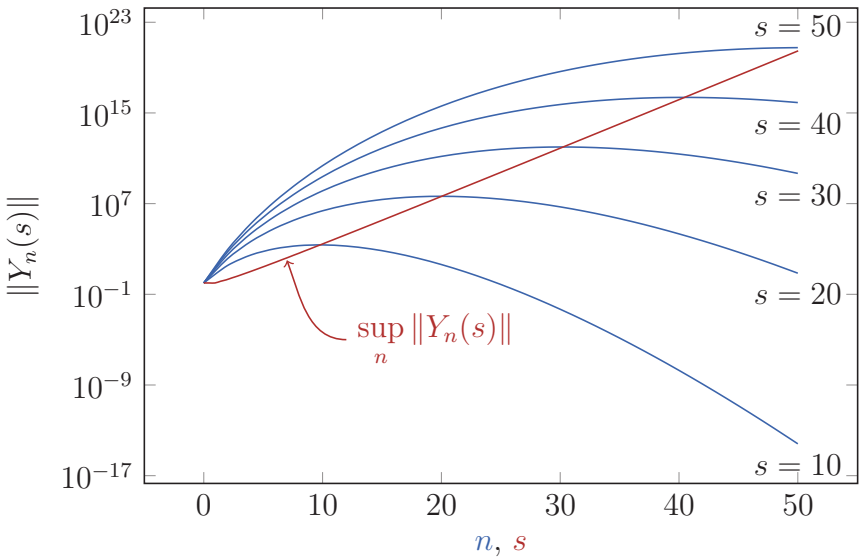


FIGURE 4.2.29 : Profil croissant puis décroissant de la norme des termes $Y_n(s)$ pour différentes valeurs de s. Noter la croissance exponentielle de $\sup_n \|Y_n(s)\|$ en fonction de s.

Récapitulons ce qui se passe dans l'algorithme $4.2.24_{48}$ pour une valeur de s « grande » (disons de l'ordre de 50). Les entrées des matrices $Y_n(s)$ sont grandes en valeur absolue (au moins pour des valeurs de n voisines de n_0) et elles sont ajoutées et retranchées entre elles à cause de l'alternance de signes de la matrice antisymétrique J — les matrices intervenant dans la récurrence $4.2.18b_{45}$ sont aussi antisymétriques, *c.f.* $4.1.5_{31}$. Tous les ingrédients de la compensation catastrophique dont il est question plus haut sont réunis et celle-ci ne manque pas de se produire dans les calculs. On pourra consulter la figure $4.2.25_{48}$ et prêter attention à la qualité de la prédiction de l'erreur due à l'estimation 4.2.28.

Synthèse Nous avons mis en évidence et expliqué le problème de compensation catastrophique sur une version simplifiée de la récurrence $4.2.18b_{45}$. C'est le fait qu'au cours de la sommation on rencontre des termes très grands (en norme) qui sont ajoutés et retranchés entre eux (du fait de la multiplication par une matrice antisymétrique) alors

que le résultat final (la somme) est compris entre -1 et 1 (on cherche une matrice de rotation) qui se révèle être responsable de la compensation catastrophique dans les calculs numériques. Les figures $4.2.25_{48}$ et $4.2.29_{49}$ ont exactement les mêmes profils dans le cas plus général de la récurrence $4.2.18b_{45}$ — les estimations d'erreurs sont aussi très bonnes — et les problèmes de compensation catastrophique sont identiques.

4.2.2.2 Notre algorithme de découpe adaptative

Pour éviter les problèmes mis en évidence dans la section précédente, on pourrait penser à augmenter la plage de calcul flottant avec des bibliothèques de type multi-précision. D'une part, la précision nécessite d'être fixée à l'avance et ceci ne fait alors que déplacer le problème : la compensation se produira pour des valeurs de s plus grandes. D'autre part ceci réduit considérablement les performances de calcul. En effet, au lieu d'être effectués directement sur le « matériel » — sur une architecture dédiée du processeur, comme c'est le cas pour les flottants —, ces calculs sont implémentés de manière « logicielle ». Nos tests ont montré que l'utilisation d'une bibliothèque multi-précision (Granlund et the GMP development team, 2012) ne permettait pas le tracé en temps réel d'une tige de taille moyenne. Celui d'une tige longue prend quand-à-lui plus d'une minute. Ceci n'est pas acceptable quand on sait que les calculs de la géométrie servent de brique de base aux calculs beaucoup plus gourmands de la simulation dynamique.

Notre algorithme est basé sur une remarque simple, comme le suggèrent les figures $4.2.20_{46}$, $4.2.25_{48}$ et $4.2.29_{49}$: la compensation catastrophique est modérée pour des petites valeurs de s. Découper le segment $[0\,;L]$ en des segments plus petits sur lesquels les calculs sont précis va ainsi nous permettre d'éviter la compensation catastrophique. Mieux, il est possible de prédire la longueur quasi-optimale des segments de manière à ne pas multiplier inutilement les découpes. En pratique le nombre de segments nécessaires à notre méthode ne dépasse pas la vingtaine, sauf pour des éléments très courbés comme sur la figure $4.2.37_{53}$. Le reste de cette section est consacré à la présentation de notre algorithme de découpe adaptative.

Principe de notre algorithme Supposons le repère $\mathcal{F}$ connu en un point $s_0 \in]0\,;L[$, grâce à un calcul sans compensation catastrophique par exemple. En considérant l'application $u \longmapsto \kappa(s_0+u)$, on construit un nouveau problème de Darboux, identique à $4.1.13_{33}$, grâce au changement de variable $s = s_0 + u$. La solution de ce nouveau problème sur le segment $[0\,;L-s_0]$ coïncide avec celle du problème précédent sur le segment $[s_0\,;L]$. De plus, d'après le théorème $4.2.1_{40}$, la solution de ce nouveau problème est développable en série entière et les termes de la série solution satisfont le système $4.2.18_{45}$. Toutes les conditions sont donc réunies pour relancer le calcul de la géométrie grâce à la récurrence $4.2.19_{46}$, pour une valeur de u pas trop grande. Si ce calcul aboutit sans encombre, on pourra répéter l'opération jusqu'à atteindre l'extrémité L du segment. Un découpage de la tige généré par notre algorithme est donné figure $4.2.35_{52}$.

Choix du pas de découpe Puisque l'on se ramène sur chaque segment à l'étude d'un problème de Darboux $4.1.13_{33}$, on peut sans perte de généralité considérer que $s_0 = 0$ et que l'on cherche $s_{\max}(s_0) > 0$ le plus grand possible tel que le calcul de $\mathcal{R}(s_{\max}(s_0))$ par la récurrence $4.2.19_{46}$ se passe sans compensation catastrophique. Pour que ceci soit le cas, il est nécessaire de borner les termes $\|\mathcal{R}_n(s)\|$ pour en contrôler la compensation. Par exemple, si l'on note ε la précision machine, et que l'on suppose que les termes $\|\mathcal{R}_n(s)\|$ ne dépassent pas $\sqrt{\varepsilon^{-1}}$, on pourra espérer une précision de l'ordre de $\sqrt{\varepsilon}$. On pourrait aussi les majorer par 1 et espérer une précision de l'ordre de ε mais la première proposition semble être un meilleur compromis entre le nombre de découpes à effectuer et la précision des calculs. Un majorant (un peu grossier) des termes $\|\mathcal{R}_n(s)\|$ a été donné dans la preuve du théorème $4.2.1_{40}$, il s'agit de $e^{C(s)}$. Un majorant de $C(s)$ garantissant que les termes $\|\mathcal{R}_n(s)\|$ restent majorés par $\sqrt{\varepsilon^{-1}}$ est donc

$$T = -\frac{1}{2}\ln\varepsilon = \frac{m}{2}\ln 2 \quad \text{si } \varepsilon = 2^{-m} \tag{4.2.30}$$

($m = 13$ en simple précision, $m = 52$ en double). Dans le cas de la double précision, on obtient $T = 26\ln 2 \approx 18{,}02$ (tronqué à 10^{-2}). Remarquons que le majorant des termes $\|\mathcal{R}_n(s)\|$ que nous avons utilisé, $e^{C(s)}$, n'était pas optimal ; un meilleur majorant, lui aussi donné dans la preuve du théorème $4.2.1_{40}$, est $M_s = \frac{C(s)^{\lfloor C(s)\rfloor}}{\lfloor C(s)\rfloor!}$. Son utilisation donne une borne T un peu plus large puisqu'en double précision, on obtient $T \approx 20{,}44$ (tronqué à 10^{-2}).

Pour obtenir la valeur de $s_{\max}(s_0)$, il reste maintenant à trouver le plus grand s satisfaisant l'inégalité

$$C(s) \leqslant T. \tag{4.2.31}$$

Rappelons que $C(s)$ est donné dans le théorème $4.2.1_{40}$, $C(s) = 2s(\|\lambda_0\| + s\|\lambda_1\|)$; c'est un polynôme de degré 2, ce qui nous permet de résoudre analytiquement l'équation 4.2.31 précédente. On obtient

$$s_{\max}(s_0) = \begin{cases} \frac{\sqrt{\|\lambda_0\|^2 + 2T\|\lambda_1\|} - \|\lambda_0\|}{2\|\lambda_1\|} & \text{si } \lambda_1 \neq 0 \\ \frac{T}{2\|\lambda_0\|} & \text{sinon et si } \lambda_0 \neq 0 \\ +\infty & \text{sinon.} \end{cases} \tag{4.2.32}$$

De ces équations on peut déduire une propriété conforme à l'intuition : dans les régions fortement courbées, on découpe beaucoup, alors que dans les régions « plates », on découpe peu. Ceci est bien visible sur la figure $4.2.35_{52}$. En effet, à $\lambda_1 \neq 0$ fixé, $\|\lambda_0\| \ll 1$ représente une région faiblement courbée alors que $\|\lambda_0\| \gg 1$ décrit une zone de forte courbure et ces comparaisons nous permettent d'obtenir les équivalents suivants

$$s_{\max}(s_0) \sim \frac{T}{2\|\lambda_0\|} \qquad \|\lambda_0\| \gg 1 \tag{4.2.33a}$$

$$s_{\max}(s_0) \sim \sqrt{\frac{T}{2\|\lambda_1\|}} \qquad \|\lambda_0\| \ll 1. \tag{4.2.33b}$$

Une fois la quantité $s_{\max}(s_0)$ calculée, on note $s_1 = s_0 + s_{\max}(s_0)$ et on peut appliquer la récurrence $4.2.19_{46}$ sur le segment $[s_0\,;s_1]$ sans craindre les problèmes numériques. Pour calculer la géométrie en un nouveau point $s > s_1$, on construit un nouveau problème de Darboux d'origine s_1 et on calcule $s_2 = s_1 + s_{\max}(s_1)$ et ainsi de suite. On crée ainsi une subdivision $(s_i)_{i\in[\![0\,;p]\!]}$ de $[0\,;L]$ telle que

$$s_0 = 0,\ \ s_p = L \quad \text{et} \quad s_{i+1} = s_i + s_{\max}(s_i) \quad \forall i \in [\![0\,;p-2]\!] \tag{4.2.34}$$

et pour laquelle les calculs de récurrence et de sommation des séries se déroulent sans compensation catastrophique. La subdivision est illustrée figure 4.2.35.

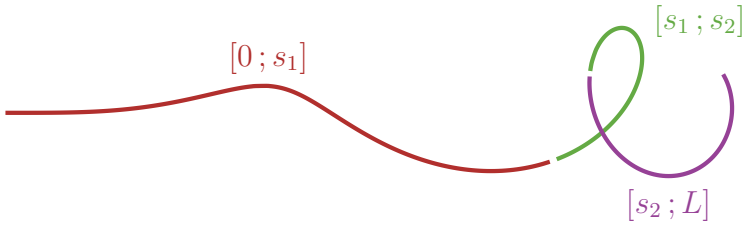


FIGURE 4.2.35 : Le découpage d'un élément de Clothoïde 3D réalisé par notre algorithme de calcul par séries entières pour éviter les erreurs numériques. On découpe plus dans les régions courbées que dans les zones « droites »[13].

Sur les figures 4.2.36 et $4.2.37_{53}$, on présente des éléments de Clothoïde 3D (un seul morceau le long duquel le vecteur de courbures varie de manière affine) dont le calcul de la géométrie échoue complètement avec l'approche naïve mais qui est très bien traité par notre algorithme.

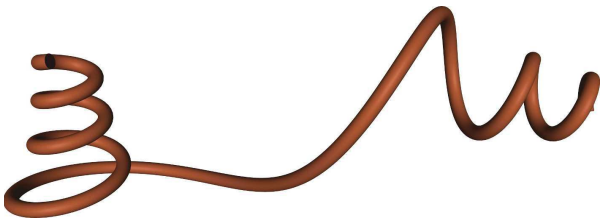


FIGURE 4.2.36 : Un élément de Clothoïde 3D dont la géométrie est calculée avec notre algorithme numériquement précis. Ceci serait impossible par l'approche naïve de l'algorithme $4.2.24_{48}$.

Avant d'aborder le raccordement des éléments de Clothoïde 3D, nous clarifions le lien entre nos travaux et ceux de Neher (1996, 1999).

[13]Ce terme n'est en fait pas très approprié car une tige droite ne subissant que de la torsion en quantité importante va être beaucoup de découpée. C'est la norme du vecteur κ qui est la quantité pertinente ici.

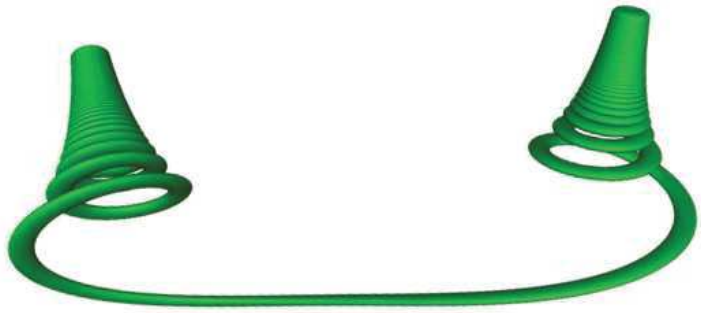


FIGURE 4.2.37 : Cas d'un élément de Clothoïde 3D, fortement courbé, calculé avec notre méthode. 109 découpes suffisent à éviter la compensation catastrophique.

Lien avec les travaux de Neher (1996, 1999) À ce stade, il nous paraît important de bien situer nos travaux par rapport à ceux de Neher (1996, 1999) que nous avons pour l'instant seulement évoqués dans la preuve du théorème $4.2.1_{40}$. Précisons d'abord que les contextes sont différents. Neher se place dans le cas scalaire mais n'exploite pas vraiment cette spécificité (comme la commutativité par exemple) et ses résultats se généralisent facilement au cas matriciel. La différence des objectifs est en fait plus importante. Le but de Neher est d'obtenir une borne sur l'erreur du calcul de la solution d'une équation différentielle par la méthode des séries entières. Ce sont donc des intervalles d'erreurs qui sont propagés pour pouvoir encadrer la somme (tronquée) de la série à la fin des calculs. Notre but n'est pas vraiment le même, une précision modeste, disons de l'ordre de $\sqrt{\varepsilon}$, est suffisante dans notre cas et l'on ne désire pas de borne exacte sur la solution approchée. Dans (Neher, 1996), un papier court de deux pages, l'auteur se consacre essentiellement aux aspects de majoration des restes de la série solution d'une équation différentielle ordinaire, de degré quelconque, linéaire à coefficients analytiques. C'est en fait la première inégalité $4.2.5_{41}$ qui est montrée mais sans que le majorant M_s soit explicité, il est remplacé par la quantité $\Gamma(n_0(s), s)$ de la preuve. Dans (Neher, 1999), l'auteur se place dans le cas d'une équation différentielle ordinaire, de degré quelconque, linéaire à coefficients polynomiaux. Un algorithme de découpe est présenté mais seulement pour seconder[14] l'utilisation de l'arithmétique flottante en multi-précision que l'auteur indique comme étant cruciale[15]. Là encore, la majoration $4.2.5_{41}$ est donnée avec $M_{\frac{s}{\omega}}$ un majorant quelconque des coefficients de la série en $\frac{s}{\omega}$. Le pas de découpe est calculé d'une manière très différente de la nôtre au moyen d'une bissection récursive de l'intervalle d'intégration. Le calcul des termes est

[14] « If the diameters of the enclosing intervals of the summands $a_k h^k$ grow due to error propagation and thus make the computation of a tight enclosure of $s(\kappa + n)$ (NdR : la somme partielle) impossible, then one must split the integration domain into subintervals, like in the case of overflow. » (Neher, 1999, p. 14).

[15] « To obtain a reasonable enclosure of the solution, i.e. a tight error bound, however, it is crucial to use a multi-precision arithmetic, as roundoff errors propagate in the recursive computation of $a_k x^k$. » (Neher, 1996, à cheval entre les pages 1 et 2). « To avoid cancellation and thus loss of accuracy in the result, the summation (6) (NdR : la somme partielle) must be carried out with multiple precision. » (Neher, 1999, p. 12).

effectué jusqu'à trouver le premier indice qui voit décroître en norme les termes de la série (le « sommet de la bosse »). Si ce sommet est trop élevé, l'intervalle est divisé en deux et l'algorithme est relancé sur chacun des sous-intervalles ainsi créés. Dans notre approche, nous trouvons la meilleure longueur (pour le majorant considéré) de pas à chaque début d'intervalle. C'est notre forme explicite du majorant $M_{\frac{s}{\omega}}$ qui est cruciale ici et nous permet de donner explicitement la valeur du pas de découpe. Notons que ceci est grandement facilité par la simplification des expressions dans notre cas d'une équation différentielle d'ordre 1. Mais notre méthode de découpage est aussi adaptée au cas polynomial, voir la section $4.2.4_{55}$. Un autre avantage de la connaissance explicite du majorant est le calcul d'une valeur ω de sorte que la majoration $4.2.5_{41}$ soit la plus fine possible. En revanche, dans (Neher, 1999), une valeur ω proche de l'optimale est calculée à chaque itération grâce à un algorithme de type Newton–Raphson. Pour conclure, on peut dire que notre approche est comparable à celle de Neher avec des optimisations qui exploitent la spécificité d'une équation différentielle d'ordre 1. Puisque notre méthode traite le cas matriciel, rien n'empêche de l'utiliser avec une équation différentielle d'ordre plus élevé en prenant soin de la mettre sous la forme d'une équation d'ordre 1 avec l'équivalence bien connue. Signalons que ce travail n'a pas été construit en utilisant celui de (Neher, 1999). En effet, cette référence nous a été indiquée a posteriori par Marc Mezzarobba dont nous reparlerons dans la section $4.3.3_{65}$. Je profite de l'occasion pour le remercier de la pertinence de cette référence.

4.2.3 Raccordement des éléments de Clothoïde 3D

Comme nous le signalions au début de cette section, les Clothoïdes 3D sont des *primitives* pour notre modèle de tige matérielle. Elles sont destinées à être raccordées entre elles, le vecteur de courbures de la courbe résultante étant une fonction affine par morceaux de l'abscisse curviligne. Pour les raisons de continuité et d'esthétique évoquées lors de la motivation de notre modèle (chapitre 3_{23}), on requiert que la courbe moyenne soit de classe $\mathscr{C}^2$. En reprenant le problème de Darboux $4.1.13_{33}$, on remarque aisément que cette propriété est satisfaite lorsque le vecteur de courbures est continu. En effet, la continuité de κ implique que la base matérielle (et donc la tangente à la courbe moyenne) est de classe $\mathscr{C}^1$. Même ci ceci n'est pas optimal, la continuité de la torsion n'étant pas nécessaire, c'est l'hypothèse que nous ferons dans la suite : le vecteur de courbures est une fonction continue et affine par morceaux de l'abscisse curviligne. Cette hypothèse a l'avantage de permettre de traiter toutes les composantes du vecteur de courbures de la même manière et de pouvoir utiliser la représentation compacte suivante. Pour décrire une telle courbe de $p \in \mathbb{N}^*$ morceaux (ou *éléments*), il suffit[16] en effet de se donner une subdivision $(s_i)_{i\in[\![0\,;p]\!]}$ de $[0\,;L]$ — nous utilisons ici la même notation que pour la subdivision de l'algorithme de découpe $4.2.34_{52}$ mais on prendra bien garde à les distinguer, l'une est numérique, l'autre est géométrique — et la valeur du vecteur de courbures aux points de la subdivision $(\kappa(s_i))_{i\in[\![0\,;p]\!]}$. Notons qu'une subdivision en p éléments confère à la tige un nombre de degrés de liberté, à un mouvement rigide près,

[16]Pour être tout à fait précis, il faudrait aussi se donner $r(0) = r_0$ et $\mathcal{R}(0) = \mathcal{R}_0$.

égal à $3(p+1)$. Nous reviendrons là-dessus lors de la comparaison avec les autres modèles de tiges dans le chapitre 5_{69}.

Un exemple de courbe moyenne d'une tige matérielle de la forme que nous considérons est donné figure 4.2.38. À titre indicatif, les découpes effectuées par notre algorithme de

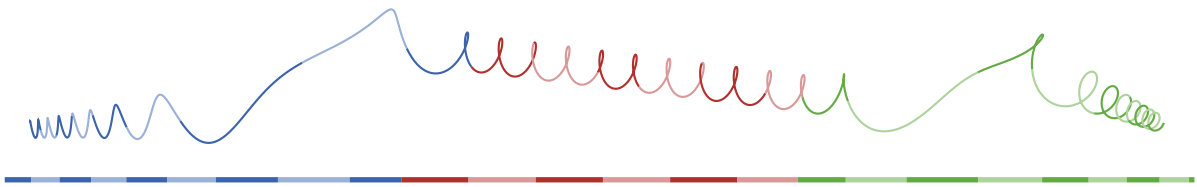


FIGURE 4.2.38 : Trois éléments de Clothoïde 3D raccordés avec une continuité $\mathscr{C}^2$. L'alternance clair/foncé indique les découpes de notre algorithme de stabilisation pour le calcul par série entière. Le découpage est plus fréquent dans les régions fortement courbées et il est uniforme pour l'élément central (en rouge) qui est une hélice circulaire[17].

calcul de la géométrie sont aussi représentées. On pourra prêter attention à la variation de leur fréquence en fonction de la courbure. Par exemple, ces découpes sont uniformes pour un élément en hélice circulaire. Il semble d'ailleurs intéressant de constater que notre algorithme utilise des segments d'un peu plus d'un tour et demi autour de l'axe de l'hélice. En effet, on peut retrouver cette propriété par le calcul en supposant pour simplifier $\kappa_2 = 0$ et $\kappa_1 > \tau > 0$ de sorte que $\|\lambda_0\| = \kappa_1$. D'après le système $4.2.32_{51}$, la longueur d'un segment dans ce cas est donnée par $\frac{T}{2\kappa_1} = \frac{\alpha}{\kappa_1}$ où α est la mesure de l'angle en radian effectué par la portion d'hélice. Si les calculs sont effectués en double précision comme c'est le cas de la figure 4.2.38, on obtient $\alpha = \frac{T}{2} \approx 10{,}22 \approx 1{,}63 \cdot 2\pi$, soit un peu plus d'un tour et demi.

La description de notre modèle géométrique est maintenant terminée. Avant d'en venir au modèle de tige dynamique auquel il permet d'aboutir, nous donnons des éléments de réflexion sur le cas d'un vecteur de courbures polynomial. En particulier nous expliquons pourquoi notre approche se généralise aux degrés supérieurs à 1.

4.2.4 Cas d'un vecteur de courbures polynomial

Jusqu'ici, nous nous sommes particulièrement intéressés au cas d'un vecteur de courbures affine (ou affine par morceaux) mais la plupart des résultats demeurent vrais dans le cas polynomial (ou polynomial par morceaux). En effet, une grande partie des majorations qui sont au cœur de notre méthode ont été dérivées dans le théorème $4.2.1_{40}$ qui se place dans le cas encore plus général d'un vecteur de courbures développable en série entière.

[17] Le calcul de la portion en hélice circulaire aurait bien sûr pu être effectué de manière directe, sans passer par les séries entières, comme dans le modèle des Super-Hélices (Bertails *et al.*, 2006) mais l'intérêt est ici pédagogique : il illustre le découpage uniforme dans les régions de courbure constante.

Dans cette section, nous étudions le cas d'un vecteur de courbures polynomial de degré d,

$$\kappa(s) = \sum_{n=0}^{d} \lambda_n s^n, \qquad \forall s \in [0\,;L]. \tag{4.2.39}$$

Nous avons vu que la quantité pertinente pour contrôler les erreurs numériques lors du calcul de la solution du problème de Darboux par la méthode des séries entières était

$$C(s) = 2\sum_{n=0}^{d} \|\lambda_n\| s^{n+1}. \tag{4.2.40}$$

En effet, cette quantité intervient non seulement dans la majoration des restes de la série (équation $4.2.5_{41}$) qui permet de contrôler l'erreur de troncature des séries (en tout cas en arithmétique exacte) mais aussi dans l'expression du pas maximum garantissant des calculs sans compensation catastrophique $4.2.31_{51}$ et $4.2.32_{51}$.

L'un des rares endroits où nous avons utilisé la spécificité du cas d'un polynôme de degré 1 (affine) était pour obtenir une expression explicite du pas de découpe dans l'équation $4.2.32_{51}$. Cependant, ceci n'était pas crucial puisque le pas de découpe apparaît comme étant l'unique zéro (si κ n'est pas identiquement nul) de la fonction $s \longmapsto C(s)-T$ sur $[0\,;L]$, qui peut facilement et efficacement être trouvé par dichotomie par exemple. En effet, $(C(0)-T)(C(L)-T) \leqslant 0$ puisque $C(0)-T = -T < 0$ et $C(L)-T \geqslant 0$, sinon c'est que la dichotomie est inutile puisque l'on peut intégrer jusqu'à L sans problème numérique.

Le plus gros problème que l'on peut alors redouter est celui d'une décroissance rapide du pas lorsque d (le degré de κ) augmente. En effet, $C(s)$ peut être élevé alors que les valeurs de $\|\kappa(s)\|$ restent modérées sur le segment $[0\,;L]$. Pour mieux comprendre ce cas, nous donnons, figure $4.2.41_{57}$, un exemple (dans le cas de la dimension 2) d'un polynôme $\kappa(s) = (0, 0, \kappa_2(s))$ dont les valeurs sur $[0\,;1]$ sont approximativement dans $[-1\,;1]$ et pour lequel le polynôme majorant $C(s)$ atteint des valeurs élevées. Ceci réside dans le fait que les coefficients de κ sont importants. En effet, le plus grand coefficient de κ est de l'ordre de $1{,}6 \cdot 10^4$ en valeur absolue et $C(1)$ est nécessairement plus grand que cette quantité.

Cependant, autour de 0, $C(s)$ est proche d'un polynôme de degré faible et l'influence des gros coefficients devant les termes de haut degré est négligeable. On remarque par exemple que pour le polynôme de la figure $4.2.41_{57}$, la condition $C(s) \leqslant T$ de l'équation $4.2.31_{51}$ qui permet de calculer $s_{\max}(0)$ donne (par lecture graphique) $s_{\max}(0) \approx 0{,}2$, ce qui laisse penser que l'algorithme va faire approximativement 4 ou 5 pas (3 ou 4 découpes). En pratique, notre algorithme de découpe continue de fonctionner remarquablement bien et les pas effectués restent raisonnables ; et ce même dans des cas extrêmes où les coefficients de κ sont de l'ordre de 10^{14} (oui, vous avez bien lu). La figure $4.2.42_{58}$ en est une illustration[18].

[18]Les profils de courbure des courbes de la figure $4.2.42_{58}$ ont été calculés par interpolation po-

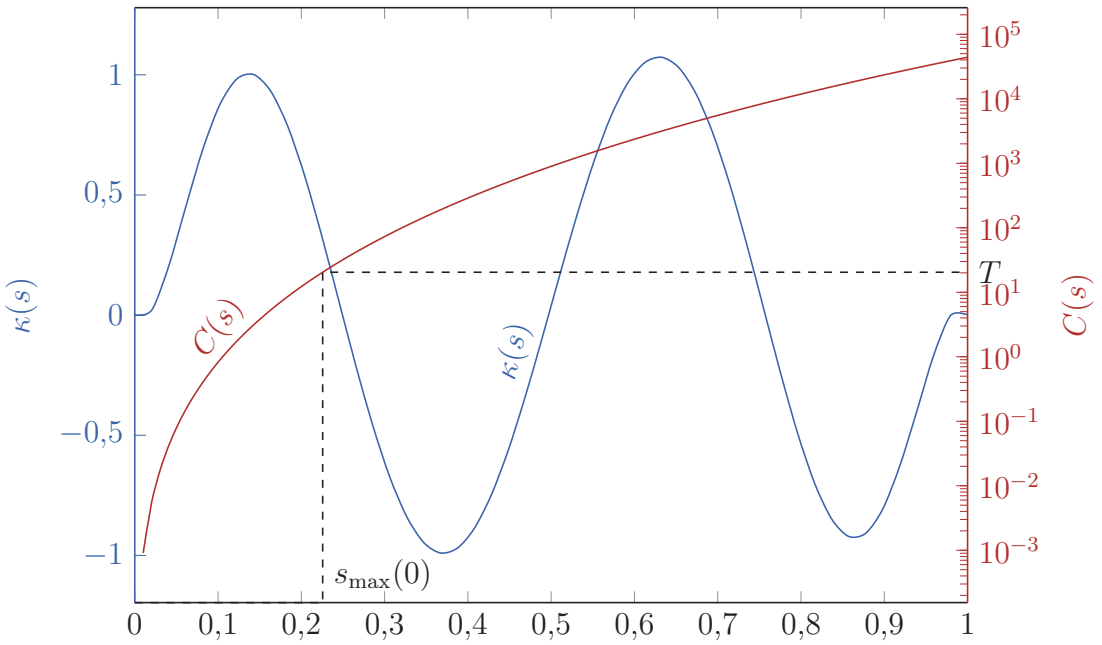


FIGURE 4.2.41 : Un exemple de polynôme $\kappa(s) = (0, 0, \kappa_2(s))$ à valeurs modérées sur $[0\,;1]$ (axe de gauche) mais pour lequel la quantité $C(s)$ prend des valeurs élevées (axe de droite). $\kappa_2(s) = -5s + 353s^2 - 3026s^3 + 10012s^4 - 15656s^5 + 11651s^6 - 3329s^7$.

Finalement, on aurait pu tout à fait utiliser des polynômes de degrés plus élevés dans nos calculs. Cependant, ceci ajoute quelques difficultés d'implémentation. Même s'il est raisonnable de considérer que le degré d du polynôme de courbures est identique sur chaque élément de la tige, afin que la répartition des degrés de liberté soit uniforme, il est préférable de ne pas imposer une continuité $\mathscr{C}^{d+1}$ de la courbe aux jonctions. Une continuité $\mathscr{C}^{d+1}$ est garantie par la continuité $\mathscr{C}^{d-1}$ du vecteur de courbures, ce qui ajoute d contraintes aux jonctions et ne laisse finalement plus que trois (un pour la torsion et deux pour les courbures) degrés de liberté par élément. En pratique, une continuité $\mathscr{C}^2$ est suffisante même pour les applications les plus sensibles (esthétique, tracé de chemin, contact). Il faudrait donc prendre en compte une telle flexibilité dans le modèle. Une autre raison de la difficulté d'implémentation ajoutée par le support du cas polynômial est celui de la simulation dynamique. Nous verrons en effet dans la section suivante que les récurrences sur les termes des séries entières nécessaires au calcul des vitesses et accélérations sont un peu plus compliquées que celles que nous avons rencontrées jusqu'ici. Les termes à propager entre les segments, ces derniers étant issus soit du découpage numérique soit du découpage géométrique, sont plus complexes surtout s'il faut traiter le cas d'un degré de continuité des courbures inférieur à $d - 1$.

---

lynomiale de $x \longmapsto A \sin\left(\frac{d\pi x}{2}\right)$ aux abscisses de Tchebychev sur $[0\,;1]$ (d est le degré du polynôme interpolant).

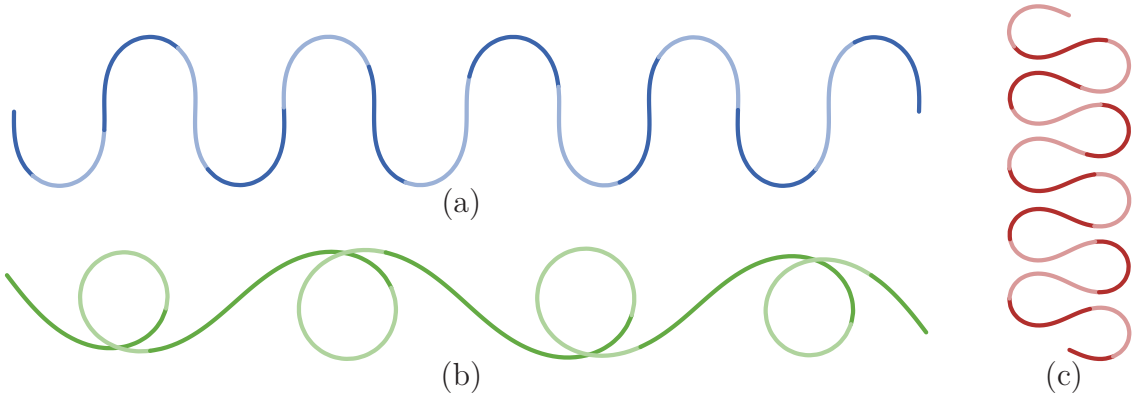


FIGURE 4.2.42 : Courbes 2D de courbures polynomiales à degrés élevés. L'alternance clair/foncé indique les découpes de l'algorithme de calcul par séries entières. Les polynômes utilisés pour (a) et (c) sont de degré 20 et ont des coefficients supérieurs à 10^{14} en valeur absolue ; celui utilisé pour (b) est de degré 8 et a des coefficients supérieurs à 10^5 en valeur absolue.

4.3 Simulation dynamique d'une Super-Clothoïde 3D

Dans cette section nous établissons les équations mécaniques qui régissent notre modèle discret de Super-Clothoïde 3D. Nous montrons notamment comment toutes les quantités cinématiques et dynamiques nécessaires à la simulation sont développables en séries entières et se calculent avec le même algorithme que celui que nous avons présenté dans la section précédente.

4.3.1 Lagrangien, principe de moindre action, équation d'Euler–Lagrange

Considérons un système mécanique conservatif, *i.e.* soumis uniquement à des forces qui dérivent d'un potentiel, dont la géométrie est complètement déterminée par la connaissance d'une fonction du temps que nous notons q, dont l'espace d'arrivée est éventuellement un espace fonctionnel. On dit que les composantes de q sont les *degrés de liberté* du système ou ses *coordonnées généralisées*.

Le *Lagrangien* du système est une fonctionnelle (une fonction qui dépend d'autres fonctions) définie par

$$\mathcal{L}(q(t), \dot{q}(t)\,; t) = E_{\mathrm{c}}(q(t), \dot{q}(t)) - V(q(t), t) \tag{4.3.1}$$

où E_{c} est l'énergie cinétique du système et V est le potentiel dont dérivent les forces en présence. Le Lagrangien permet de définir l'*action* du système entre des instants t_1 et

t_2,

$$S = \int_{t_1}^{t_2} \mathcal{L}(q(t), \dot{q}(t)\,; t)\,\mathrm{d}t. \tag{4.3.2}$$

Enfin, le *principe de moindre action* (voir par exemple (Goldstein *et al.*, 2001)) affirme que la trajectoire suivie par le système entre les instants t_1 et t_2 est telle que l'action S est minimale. C'est le principe de base de la *mécanique analytique*.

Pour aboutir à une équation différentielle sur q, considérons un petit déplacement (virtuel) des coordonnées généralisées, δq ; il implique une variation $\delta\dot{q}$ des vitesses. On suppose que ce déplacement ne modifie pas les positions du système aux instants final et initial, c'est-à-dire $\delta q(t_1) = \delta q(t_2) = 0$, de sorte que seule la trajectoire soit perturbée. Regardons quelle variation δS ceci implique sur l'action du système.

$$\delta S = \int_{t_1}^{t_2} \left(\left\langle \frac{\partial \mathcal{L}}{\partial q}\,; \delta q \right\rangle + \left\langle \frac{\partial \mathcal{L}}{\partial \dot{q}}\,; \delta \dot{q} \right\rangle \right) \mathrm{d}t \tag{4.3.3a}$$

$$= \int_{t_1}^{t_2} \left(\left\langle \frac{\partial \mathcal{L}}{\partial q}\,; \delta q \right\rangle - \left\langle \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathcal{L}}{\partial \dot{q}} \right)\,; \delta q \right\rangle \right) \mathrm{d}t \tag{4.3.3b}$$

$$= \int_{t_1}^{t_2} \left\langle \frac{\partial \mathcal{L}}{\partial q} - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathcal{L}}{\partial \dot{q}} \right)\,; \delta q \right\rangle \mathrm{d}t. \tag{4.3.3c}$$

L'action étant minimum, δS doit être nul pour toute variation admissible δq ; grâce au *lemme fondamental du calcul des variations*, on obtient l'*équation d'Euler–Lagrange*,

$$\frac{\partial \mathcal{L}}{\partial q} - \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathcal{L}}{\partial \dot{q}} \right) = 0. \tag{4.3.4}$$

Notons que le second terme comporte une dérivation « totale » par rapport au temps (et non partielle) c'est-à-dire qu'il ne faut pas seulement tenir compte de la variation par rapport au temps du Lagrangien, mais aussi de celle de q et de $\dot{q}$.

Application au cas d'une tige de Kirchhoff

Le lecteur aura bien compris que c'est le vecteur de courbures qui va jouer le rôle des coordonnées généralisées dans notre cas. Nous avons en effet montré dans la section $4.1.3_{33}$ que c'était une quantité pertinente pour décrire toute la géométrie d'une tige de Kirchhoff ; au moins dans le cas où celle-ci est encastrée en $s = 0$, *i.e.* dans le cas où r_0 et $\mathcal{R}_0$ ne sont pas des degrés de liberté du système. Dans la suite, on note $q = \kappa$.

Le Lagrangien de notre tige de Kirchhoff est donné par

$$\mathcal{L}(q, \dot{q}) = E_{\mathrm{c}}(q, \dot{q}) - E_{\mathrm{int}}(q) - E_{\mathrm{ext}}(q) \tag{4.3.5}$$

avec

$$E_{\mathrm{c}}(q,\dot{q}) = \frac{1}{2}\int_0^L \|\dot{r}\|^2 \rho S \,\mathrm{d}s \tag{4.3.6a}$$

$$E_{\mathrm{int}}(q) = \frac{1}{2}\int_0^L \|q-\bar{q}\|_{\mathbb{K}}^2 \,\mathrm{d}s \qquad \{\text{équation } 4.1.38_{39}\} \tag{4.3.6b}$$

$$E_{\mathrm{ext}}(q) = -\int_0^L \langle r\,;p\rangle \,\mathrm{d}s \tag{4.3.6c}$$

où p est la densité linéique de force extérieure appliquée à la tige que nous avions déjà introduite pour établir les équations de Kirchhoff $4.1.33_{38}$, elle est ici supposée conservative. Pour un élément h de l'espace des coordonnées généralisées, les quantités intervenant dans l'équation d'Euler–Lagrange s'écrivent,

$$\left\langle \frac{\partial E_{\mathrm{c}}}{\partial q}\,;h\right\rangle = \int_0^L \left\langle \dot{r}\,;D^2r(\dot{q},h)\right\rangle \rho S \,\mathrm{d}s \tag{4.3.7a}$$

$$\left\langle \frac{\partial E_{\mathrm{c}}}{\partial \dot{q}}\,;h\right\rangle = \int_0^L \langle \dot{r}\,;Dr\,h\rangle \rho S \,\mathrm{d}s \tag{4.3.7b}$$

$$\left\langle \frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial E_{\mathrm{c}}}{\partial \dot{q}}\,;h\right\rangle = \int_0^L \left(\langle \ddot{r}\,;Dr\,h\rangle + \left\langle \dot{r}\,;D^2r(\dot{q},h)\right\rangle\right) \rho S \,\mathrm{d}s \tag{4.3.7c}$$

$$\left\langle \frac{\partial E_{\mathrm{int}}}{\partial q}\,;h\right\rangle = \int_0^L \langle \mathbb{K}(q-\bar{q})\,;h\rangle \,\mathrm{d}s \tag{4.3.7d}$$

$$\left\langle \frac{\partial E_{\mathrm{ext}}}{\partial q}\,;h\right\rangle = \int_0^L \langle Dr\,h\,;-p\rangle \,\mathrm{d}s. \tag{4.3.7e}$$

L'équation d'Euler–Lagrange projetée sur h s'écrit alors

$$\int_0^L \left\langle [Dr]^\top(\rho S\,\ddot{r} - p) + \mathbb{K}(q-\bar{q})\,;h\right\rangle \mathrm{d}s = 0 \tag{4.3.8}$$

où

$$\ddot{r} = Dr\,\ddot{q} + D^2r(\dot{q},\dot{q}). \tag{4.3.9}$$

Rappelons que ces résultats sont généraux et s'appliquent au vecteur de courbures non discrétisé en espace. Nous appliquons maintenant ces résultats au cas d'une tige en Clothoïdes 3D par morceaux (la courbure est un polynôme de degré 1 par morceaux de l'abscisse curviligne).

4.3.2 Application au calcul de la dynamique d'une Super-Clothoïde 3D

Nous avons déjà vu qu'il n'est pas possible de trouver une solution générique de forme fermée (voir l'annexe A_{157} pour une définition) aux équations de Kirchhoff $4.1.33_{38}$. Pour en donner une solution approchée, nous avons limité l'espace admissible du vecteur de

courbures en imposant qu'il soit affine par morceaux. On dit que l'on cherche une *solution faible* des équations de Kirchhoff[19].

Le grand avantage de la formulation Lagrangienne que nous avons développée ci-dessus est de s'appliquer dans le cas continu comme dans le cas discret. En effet, si l'on note maintenant q un vecteur contenant les coordonnées discrètes de notre modèle — les courbures aux nœuds par exemple, voir plus loin —, on construit un Lagrangien discret de la même manière qu'en $4.3.5_{59}$ pour lequel on peut écrire l'équation d'Euler–Lagrange qui sera très proche de l'équation continue $4.3.8_{60}$. Le vecteur q ne dépend maintenant plus que du temps (on a « éliminé » la variable d'espace) et les équations $4.3.8_{60}$ et $4.3.9_{60}$ peuvent se réécrire un peu plus simplement,

$$\mathbb{M}(q)\,\ddot{q} + \underline{\mathbb{K}}(q - \bar{q}) = F(q, \dot{q}) \tag{4.3.10}$$

où

$$\mathbb{M}(q) = \int_0^L [Dr]^\top Dr\, \rho S \,\mathrm{d}s \tag{4.3.11a}$$

$$F(q, \dot{q}) = \int_0^L [Dr]^\top p \,\mathrm{d}s - \int_0^L [Dr]^\top D^2 r(\dot{q}, \dot{q})\, \rho S \,\mathrm{d}s. \tag{4.3.11b}$$

Pour préciser le contenu de la matrice $\underline{\mathbb{K}}$, nous devons préciser celui du vecteur q. On suppose que la tige est constituée de p éléments de Clothoïde 3D numérotés de 1 à p dans le sens des abscisses curvilignes croissantes. Chaque élément d'indice i est de longueur $\ell_i > 0$ et compris entre les abscisses s_{i-1} et s_i de sorte que

$$s_0 = 0, \quad s_p = L \quad \text{et} \quad s_i = s_{i-1} + \ell_i, \quad \forall i \in [\![1\,;p]\!]. \tag{4.3.12}$$

Le vecteur q est alors un élément de $\mathbb{R}^{3(p+1)}$ qui contient les courbures aux *nœuds* de la subdivision $(s_i)_{i\in[\![0\,;p]\!]}$,

$$q_i = \kappa(s_{i-1}), \quad \forall i \in [\![1\,;p+1]\!] \tag{4.3.13}$$

et il permet d'exprimer la courbure en tout point par la relation

$$\kappa(s) = \sum_{i=1}^{p} \left(\frac{s_i - s}{\ell_i}\, q_i + \frac{s - s_{i-1}}{\ell_i}\, q_{i+1} \right) \chi_{[s_{i-1};s_i[}(s) \tag{4.3.14a}$$

$$= \sum_{i=1}^{p+1} \left(\frac{s_i - s}{\ell_i} \chi_{[s_{i-1};s_i[}(s) + \frac{s - s_{i-2}}{\ell_{i-1}} \chi_{[s_{i-2};s_{i-1}[}(s) \right) q_i \tag{4.3.14b}$$

$$= \sum_{i=1}^{p+1} \Lambda_i(s)\, q_i, \qquad \forall s \in [0\,;L[\tag{4.3.14c}$$

[19] Ceci revient à supposer que le déplacement virtuel $\delta\kappa$ que nous avons considéré pour établir l'expression de l'énergie interne de la tige dans la section $4.1.4_{35}$ n'est plus quelconque mais d'une forme particulière donnée. Dans notre cas, cette forme est en fait explicitée en 4.3.14c.

où $\chi_{[a;b[}$ est la fonction indicatrice sur $[a\,;b[$ — elle vaut 1 sur $[a\,;b[$ et zéro partout ailleurs — et où on a pris comme convention $s_{-1} = s_0$ et $s_{p+1} = s_p$. Le vecteur q apparaît comme étant la décomposition de la fonction κ sur la base de fonctions $\Lambda = (\Lambda_i)_{i\in[\![1\,;p+1]\!]}$. On a ainsi restreint l'espace des solutions (faibles) des équations de Kirchhoff à celui engendré par Λ. Pour un aperçu des fonctions Λ_i, on pourra se reporter à la figure 4.3.15 et prêter notamment attention à l'influence d'une coordonnée q_i, localisée sur $]s_{i-2}\,;s_i[$.

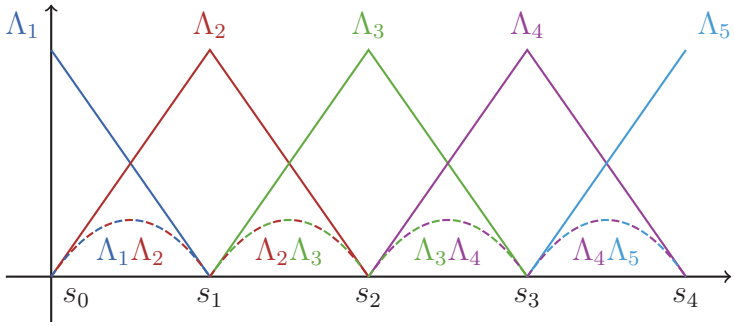


FIGURE 4.3.15 : Les fonctions de forme qui régissent notre discrétisation spatiale 4.3.14c$_{61}$. On remarque que l'influence d'une coordonnée q_i est localisée sur la portion de courbe $]s_{i-2}\,;s_i[$. On a aussi tracé les fonctions produit intervenant dans les calculs de $\underline{\mathbb{K}}$, *c.f.* équation 4.3.18.

Donnons maintenant l'expression de la matrice $\underline{\mathbb{K}}$. Le produit scalaire qui lui est associé dans $\mathbb{R}^{3(p+1)}$ apparaît comme l'homologue de celui associé à $\mathbb{K}$ dans $\mathscr{C}([0\,;L],\mathbb{R}^3)$. En effet, pour que 4.3.8$_{60}$ et 4.3.10$_{61}$ soient équivalentes, $\underline{\mathbb{K}}$ et $\mathbb{K}$ doivent définir les mêmes produits scalaires sur leur espace respectif. Plus précisément, pour $\underline{h}$ et $\underline{h}'$ deux éléments de $\mathbb{R}^{3(p+1)}$, on note h et h' leur fonction de courbure respective donnée par l'équation 4.3.14c$_{61}$, on a nécessairement,

$$\langle h\,;h'\rangle_{\mathbb{K}} = \langle \underline{h}\,;\underline{h}'\rangle_{\underline{\mathbb{K}}} \Longleftrightarrow \langle \mathbb{K}\,h\,;h'\rangle = \langle \underline{\mathbb{K}}\,\underline{h}\,;\underline{h}'\rangle. \tag{4.3.16}$$

En développant ceci sur la base Λ, on obtient,

$$\langle \underline{\mathbb{K}}\,\underline{h}\,;\underline{h}'\rangle = \int_0^L \left\langle \mathbb{K}\sum_{i=1}^{p+1}\Lambda_i\,\underline{h}_i\,;\sum_{j=1}^{p+1}\Lambda_j\,\underline{h}'_j\right\rangle \mathrm{d}s = \sum_{i=1}^{p+1}\sum_{j=1}^{p+1}\int_0^L \langle \mathbb{K}\underline{h}_i\,;\underline{h}'_j\rangle\,\Lambda_i\,\Lambda_j\,\mathrm{d}s \tag{4.3.17a}$$

$$= \sum_{i=1}^{p+1}\sum_{j=1}^{p+1}\langle \mathbb{K}\underline{h}_i\,;\underline{h}'_j\rangle \int_0^L \Lambda_i\,\Lambda_j\,\mathrm{d}s. \tag{4.3.17b}$$

De cette dernière équation on déduit que le bloc 3×3 d'indice i,j de la matrice $\underline{\mathbb{K}}$, $\underline{\mathbb{K}}_{ij}$, est donné par

$$\underline{\mathbb{K}}_{ij} = \mathbb{K}\int_0^L \Lambda_i(s)\,\Lambda_j(s)\,\mathrm{d}s. \tag{4.3.18}$$

Les fonctions $\Lambda_i\Lambda_j$ sont représentées figure 4.3.15$_{62}$. Finalement, la matrice $\underline{\mathbb{K}}$ s'écrit

$$\underline{\mathbb{K}} = \begin{pmatrix} \frac{\ell_1}{3}\mathbb{K} & \frac{\ell_1}{6}\mathbb{K} & 0 & \cdots & 0 \\ \frac{\ell_1}{6}\mathbb{K} & \frac{\ell_1+\ell_2}{3}\mathbb{K} & \frac{\ell_2}{6}\mathbb{K} & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \frac{\ell_{p-1}}{6}\mathbb{K} & \frac{\ell_{p-1}+\ell_p}{3}\mathbb{K} & \frac{\ell_p}{6}\mathbb{K} \\ 0 & \cdots & 0 & \frac{\ell_p}{6}\mathbb{K} & \frac{\ell_p}{3}\mathbb{K} \end{pmatrix}, \tag{4.3.19}$$

elle est tridiagonale par blocs.

Remarquons que nous avons transformé le système d'équations aux dérivées partielles de Kirchhoff 4.1.33$_{38}$ en une équation différentielle ordinaire dont la variable représente une approximation en espace. Ce genre d'approche est commune à la plupart des méthodes de résolution d'équations aux dérivées partielles (différences finies, éléments finis, *etc.*).

On dispose maintenant de l'expression de toutes les quantités nécessaires à la construction du système 4.3.10$_{61}$. Dans la section suivante, on montre comment ces quantités peuvent être calculées numériquement sous forme de séries entières.

Remarque 4.3.20. Signalons ici que, comme dans (Bertails *et al.*, 2006), on peut ajouter un terme dissipatif empirique de frottement interne, en

$$\frac{1}{2}\int_0^L \nu\|\dot{\kappa}\|_{\mathbb{K}}^2 \,\mathrm{d}s. \tag{4.3.21}$$

Une fois discrétisé, ce terme contribue au membre de gauche de l'équation 4.3.10$_{61}$ via la quantité

$$\nu\,\underline{\mathbb{K}}\,\dot{q}. \tag{4.3.22}$$

On peut encore compléter le modèle par une force externe de frottement visqueux du type

$$-\,\alpha\,\dot{r} \tag{4.3.23}$$

qui modifie le membre de droite de l'équation 4.3.10$_{61}$ en ajoutant le terme

$$-\,\alpha\int_0^L Dr^\top\dot{r}\,\mathrm{d}s = -\frac{\alpha}{\rho S}\mathbb{M}(q)\,\dot{q}. \tag{4.3.24}$$

Remarque 4.3.25. Signalons aussi que l'hypothèse d'encastrement peut être relâchée. En effet, il suffit pour cela de considérer que $\mathcal{R}_0$ et r_0 sont des degrés de liberté du système. Il deviennent des éléments de la coordonnée généralisée q. On est alors amené à dériver le Lagrangien par rapport à ces paramètres et leur calcul par série entière ne pose finalement pas de problème.

4.3.3 Calcul des quantités dynamiques par séries entières

L'un des points clef du succès de l'approche par séries entières pour la simulation dynamique est que *tous* les termes de l'équation $4.3.10_{61}$ sont développables en séries entières et calculables numériquement par la méthode que nous avons utilisée pour le calcul de la géométrie.

En effet, d'après la théorie des équations différentielles à paramètres (Demailly, 2006, théorème p. 302), la solution du problème de Darboux $4.1.13_{33}$ est de classe $\mathscr{C}^\infty$ par rapport au paramètre q. De plus, les différentielles successives $D^n\mathcal{F}$ de $\mathcal{F}$ sont solutions des équations différentielles obtenues en dérivant n fois le problème de Darboux par rapport à la variable q. Ce dernier étant linéaire par rapport à cette variable, ses problèmes dérivés le sont aussi et par les mêmes arguments qu'au théorème $4.2.1_{40}$ on montre que les solutions (les $D^n\mathcal{F}$) sont développables en séries entières et leur termes généraux satisfont des équations de récurrence qui permettent de les calculer de proche en proche.

Afin d'expliciter ces équations de récurrence, on peut faire comme lors de la démonstration du théorème $4.2.1_{40}$, c'est-à-dire « ré-injecter » le développement dans l'équation différentielle, mais il y a plus simple. En effet, il suffit pour cela de dériver les récurrences déjà établies.

Prenons le cas d'un seul élément de Clothoïde 3D ($p = 1$), nous verrons plus loin comment on l'étend facilement au cas d'une Super-Clothoïde 3D. On a $\kappa(s) = \lambda_0 + s\lambda_1 = \frac{L-s}{L}\, q_1 + \frac{s}{L}\, q_2$, *i.e.* $\lambda_0 = q_1$ et $\lambda_1 = \frac{q_2 - q_1}{L}$. La récurrence qui régit les termes de la série $D\mathcal{F}$ s'obtient en différentiant $4.2.18_{45}$. $\forall n \in \mathbb{N}$, $\forall h \in \mathbb{R}^{3(p+1)}$,

$$\begin{cases} D\mathcal{R}_0 = 0 \\ D\mathcal{R}_1 = \mathcal{R}_0 D[\lambda_0]_\times \\ D\mathcal{R}_{n+2}\, h = \dfrac{1}{n+2}\left(D\mathcal{R}_{n+1}\, h\, [\lambda_0]_\times + \mathcal{R}_{n+1} D[\lambda_0]_\times h + D\mathcal{R}_n\, h\, [\lambda_1]_\times + \mathcal{R}_n D[\lambda_1]_\times h\right) \\ Dr_0 = 0 \\ Dr_{n+1} = \dfrac{1}{n+1} Dt_n. \end{cases}$$

À l'ordre 2 on trouve, $\forall n \in \mathbb{N}$, $\forall h, h' \in \mathbb{R}^{3(p+1)}$,

$$\begin{cases} D^2\mathcal{R}_0 = 0 \\ D^2\mathcal{R}_1 = 0 \\ D^2\mathcal{R}_{n+2}(h,h') = \dfrac{1}{n+2}\Big(D^2\mathcal{R}_{n+1}(h,h')[\lambda_0]_\times + D\mathcal{R}_{n+1}\, hD[\lambda_0]_\times h' + D\mathcal{R}_{n+1}\, h'D[\lambda_0]_\times h \\ \qquad\qquad + D^2\mathcal{R}_n(h,h')[\lambda_1]_\times + D\mathcal{R}_n\, hD[\lambda_1]_\times h' + D\mathcal{R}_n\, h'D[\lambda_1]_\times h\Big) \\ D^2 r_0 = 0 \\ D^2 r_{n+1} = \dfrac{1}{n+1} D^2 t_n. \end{cases}$$

On remarque que si ces deux récurrences sont effectuées en même temps que celles du repère $\mathcal{F}$, toutes les quantités nécessaires sont connues et le calcul peut effectivement être réalisé.

On sait donc calculer par exemple le premier terme de 4.3.11b$_{61}$ dans le cas où la force extérieure p est constante — c'est le cas pour la force de gravité puisqu'elle correspond à $p = \rho S\, g$ —, l'intégration par rapport à s étant triviale avec les séries entières. C'est encore plus simple dans le cas où p est une distribution de Dirac — comme pour des forces de contact discrètes — puisqu'il n'y a pas d'intégration spatiale. On remarque que tous les autres termes ($\mathbb{M}$ et le second terme de F) font intervenir un produit. Le produit de deux séries est un *produit de Cauchy* et nous nous attachons maintenant à leur traitement.

Le cas des produits de Cauchy

Si deux séries entières $\sum a_n s^n$ et $\sum b_n s^n$ dans une algèbre de Banach — penser : un espace complet qui possède une loi multiplicative — convergent absolument, alors leur produit est aussi développable en séries entière et vérifie

$$\left(\sum_{n=0}^{+\infty} a_n s^n\right)\left(\sum_{n=0}^{+\infty} b_n s^n\right) = \sum_{n=0}^{+\infty}\left(\sum_{k=0}^{n} a_k\, b_{n-k}\right) s^n. \tag{4.3.28}$$

Pour nous, ce résultat présente à la fois un avantage et un inconvénient. D'une part il garantit que l'on va pouvoir continuer à calculer nos quantités avec des séries entières mais d'autre part il montre que ces calculs vont être coûteux. En effet, la complexité du calcul du n-ième terme de la série produit est en $\mathcal{O}(n)$. Nous verrons cependant dans le chapitre 5$_{69}$ que ceci n'est pas dramatique pour le temps de calcul. Ces produits représentent toutefois une part importante de la charge de calcul (72% pour une simulation raisonnable) ce qui nous a conduit à chercher un moyen de les éviter. Ces réflexions ont été menées conjointement avec Marc Mezzarobba et Bruno Salvy, deux spécialistes du calcul formel au CNRS et à l'Inria. Une approche légitime pour éviter ces produits de Cauchy est de chercher une équation différentielle à coefficients polynomiaux satisfaite par le produit. Il semble que ceci soit possible au prix d'une récurrence d'ordre assez élevé et de la présence de « pôles ». En effet, l'équation trouvée n'est pas *résolue*, c'est-à-dire que le coefficient devant la dérivée de plus haut degré n'est pas 1 mais un polynôme avec des racines réelles, les « pôles ». Il n'est pas sûr que ces pôles posent problème et on pourrait encore espérer trouver une version « désingularisée » (sans pôle) de l'équation, au prix de faire encore gonfler l'ordre de la récurrence. Le gain possible en efficacité semble finalement relativement limité. À ce stade de la réflexion, il nous a semblé préférable de ne pas persister dans la recherche abusive de performance, ceci n'étant pas la priorité de cette thèse. Cette piste de réflexion reste cependant intéressante et ouverte.

Adaptation de l'algorithme de découpe

Nous sommes maintenant en mesure de calculer, au moins en arithmétique exacte, tous les termes de l'équation dynamique 4.3.10$_{61}$. Cependant, rien ne garantit que les termes

des séries en jeu ne vont pas subir le genre de problème que nous nous sommes appliqués à éviter dans la section $4.2.2_{45}$: la compensation catastrophique. Nous allons montrer que, hormis une modification du pas d'intégration, notre algorithme peut être appliqué directement.

La différentielle $D\mathcal{R}$ est solution d'une équation différentielle linéaire à coefficients polynomiaux, d'ordre 1 avec un second membre développable en série entière. Ce type d'équation est proche de celle du théorème $4.2.1_{40}$ au second membre près. On peut en fait montrer que les résultats du théorème $4.2.1_{40}$ subsistent : les restes de la série sont majorés par une suite géométrique dont la raison peut être choisie arbitrairement petite. On pourrait dériver une borne précise des coefficients de la série afin d'obtenir une valeur optimale du pas de découpe mais, cela n'est en fait pas utile. En effet, on montre que le pas optimal de découpe est dans ce cas toujours plus grand que celui nécessaire au calcul des produits de Cauchy (voir le paragraphe suivant). Tous les termes des différentes séries étant calculés simultanément, le découpage adopté est celui de la série la plus contraignante ; le calcul de $D\mathcal{R}$ (et donc celui de Dr) ne nécessite pas de découpage supplémentaire. Le même raisonnement peut être effectué pour les différentielles secondes $D^2\mathcal{R}$ et D^2r puisque ce sont aussi des solutions d'équations différentielles avec second membre, du même type que précédemment.

Attachons-nous maintenant au cas des produits de Cauchy. Remarquons que dans les termes de l'équation $4.3.10_{61}$, tous les produits de Cauchy sont intégrés et souvenons-nous que l'important pour éviter la compensation catastrophique est de contrôler la norme des termes de la série. Si l'on note $(c_n s^n)_{n\in\mathbb{N}}$ le terme général de la série *primitive* du produit du paragraphe précédent, on a la majoration suivante,

$$\left\|c_{n+1}s^{n+1}\right\| \overset{20}{\leqslant} \frac{s}{n+1}\sum_{k=0}^{n}\left\|a_k s^k\right\|\left\|b_{n-k}s^{n-k}\right\| \leqslant \frac{s}{n+1}\sum_{k=0}^{n} M_s^a M_s^b \leqslant \overbrace{sM_s^a M_s^b}^{M_s^c} \tag{4.3.29}$$

où M_s^a et M_s^a sont respectivement des majorants des termes $(a_n s^n)_{n\in\mathbb{N}}$ et $(b_n s^n)_{n\in\mathbb{N}}$. Cette dernière majoration est importante car elle assure la décroissance rapide des termes de la série primitive du produit. En effet, elle permet de donner une majoration des restes de la série par une suite géométrique semblable à celle du théorème $4.2.1_{40}$, puisque pour $n \in \mathbb{N}$ et $\omega \in \,]0\,;1[$,

$$\begin{aligned}\left\|\sum_{k=n}^{+\infty} c_{k+1}s^{k+1}\right\| &\overset{21}{\leqslant} \sum_{k=n}^{+\infty}\frac{s}{k+1}\omega^k\sum_{i=0}^{k}\left\|a_i\left(\frac{s}{\omega}\right)^i\right\|\left\|b_{k-i}\left(\frac{s}{\omega}\right)^{k-i}\right\| \leqslant \sum_{k=n}^{+\infty} sM_{\frac{s}{\omega}}^a M_{\frac{s}{\omega}}^b\,\omega^k \\ &\leqslant sM_{\frac{s}{\omega}}^a M_{\frac{s}{\omega}}^b\,\frac{\omega^n}{1-\omega} \leqslant M_{\frac{s}{\omega}}^c\,\frac{\omega^{n+1}}{1-\omega}.\end{aligned} \tag{4.3.30}$$

Pour garantir que M_s^c ne soit pas trop élevé et ainsi éviter la compensation catastrophique, il suffit que sM_s^{a2} et sM_s^{b2} ne dépassent pas le seuil précédemment fixé, $\sqrt{\varepsilon^{-1}}$,

[20] Ici nous n'avons pas précisé dans quel espace vivaient nos suites et il est possible qu'une constante apparaisse dans la majoration de la norme du produit par le produit des normes. Pour en tenir compte, il suffit d'ajouter cette constante dans la quantité M_s^c.

[21] Voir note 20.

c.f. section $4.2.2.2_{50}$. On peut alors dériver le même genre de borne sur s que celle de l'équation $4.2.31_{51}$. En pratique, ceci revient à couper environ deux fois plus[22]. On pourrait croire que cela a un impact négatif sur le temps de calcul mais ceci a aussi pour effet de diminuer — environ par 2 — le nombre de termes requis sur chaque segment et n'est en fait pas très coûteux. C'est d'ailleurs un point important pour comprendre pourquoi le calcul des produits de Cauchy n'est pas rédhibitoire : en limitant le nombre de termes des séries dont on veut faire le produit, on limite aussi le nombre de termes dans la somme du produit de Cauchy $4.3.28_{65}$.

Cas d'une Super-Clothoïde 3D

Jusqu'ici nous n'avons considéré qu'un seul élément de Clothoïde 3D. Au paragraphe précédent, nous avons déjà donné les modifications à apporter pour passer d'un segment de découpe numérique au suivant. Le passage entre segments de découpe numérique et celui entre segments géométriques (éléments) est en fait identique. Seule l'information de courbure change mais les termes d'initialisation des récurrences sont simplement ceux du segment précédent.

Conclusion

Dans cette section nous avons montré que notre méthode de résolution du problème de Darboux (géométrique) s'étendait naturellement aux calculs des quantités dynamiques. Avant de présenter nos résultats et de nous comparer aux modèles de référence, nous présentons notre schéma d'intégration en temps.

4.3.4 Intégration en temps

Maintenant que l'on sait complètement et précisément calculer les termes de l'équation dynamique $4.3.10_{61}$, il s'agit de résoudre, au moins de manière approchée, cette équation. Entre stabilité, précision, et efficacité, c'est toujours la stabilité et l'efficacité qui priment en informatique graphique. La stabilité est assurée par une résolution *implicite* de l'équation dynamique ; c'est-à-dire que les quantités à l'instant $n+1$ sont définies de manière implicite par une équation du type $x_{n+1} = F(x_n, x_{n+1})$. Dans le cas d'une équation non-linéaire comme la nôtre, ces schémas requièrent une recherche de zéro, souvent effectuée grâce à un algorithme de type Newton–Raphson. Le fait que dans le cas présent la matrice $\mathbb{M}$ dépende non-linéairement de q et soit « pleine », va considérablement ralentir ce genre de méthode. C'est pourquoi, afin de ne pas trop dégrader l'efficacité de la simulation, nous utilisons un schéma numérique implicite-explicite où certains termes sont explicites et d'autres implicites[23]. Il est bon de rappeler ici que l'une des motivations qui nous a conduit à considérer le vecteur de courbures comme degré de

[22] Pour voir ceci « avec les mains », on peut reprendre l'équation $4.2.31_{51}$ en se souvenant que $C(s) \approx \ln(M_s)$ et donc $\ln(s{M_s}^2) \approx 2C(s) + \ln s$. Le terme $C(s)$ étant polynomial, il domine $\ln s$.

[23] D'une manière générale, la terminologie « semi-implicite » est ambiguë. Elle peut faire référence à un schéma où certains termes sont explicites, d'autres implicites, comme celui que nous utilisons. Elle peut aussi désigner un schéma de type *Euler symplectique* où on intègre la vitesse explicitement puis

liberté de notre modèle est que la force élastique en dépend linéairement (*c.f.* $4.3.10_{61}$). Ceci permet l'implicitation aisée des termes élastiques, ce qui est bienvenu car ce sont les termes les plus « raides » de l'équation $4.3.10_{61}$; les autres termes seront explicités. Notre schéma s'écrit alors

$$\begin{cases} \left(\mathbb{M}(q_k) + \Delta t^2 \underline{\mathbb{K}}\right) \Delta \dot{q}_k = \Delta t \left(F(q_k, \dot{q}_k) - \underline{\mathbb{K}}(q_k + \Delta t\, \dot{q}_k - \bar{q})\right) & \text{(4.3.31a)} \\ \Delta q_k = \Delta t(\dot{q}_k + \Delta \dot{q}_k) & \text{(4.3.31b)} \end{cases}$$

où Δt est le pas d'intégration, q_k et $\dot{q}_k$ sont les approximations de $q(t_k)$ et $\dot{q}(t_k)$ à l'instant $t_k = k\Delta t$ et $\Delta q_k = q_{k+1} - q_k$ et $\Delta \dot{q}_k = \dot{q}_{k+1} - \dot{q}_k$ sont les incréments en position et en vitesse au pas k.

Ce schéma est identique à celui utilisé dans (Bertails *et al.*, 2006). Il est d'ordre 1, et malgré le peu d'implicitation qu'il comporte, il est étonnamment stable. Son succès réside dans le fait que ce sont les termes les plus « raides » (les termes élastiques) qui ont été implicités.

La mise en œuvre de ce schéma d'intégration ainsi que la comparaison de notre modèle avec des standards de la littérature sont données au chapitre suivant.

Remarque 4.3.32. On aurait aussi pu impliciter au premier ordre les termes en $\dot{q}$ dans F au prix de rompre la symétrie de la matrice du membre de gauche de 4.3.31a. Nous avons testé cette alternative et constaté expérimentalement que le schéma était un peu plus stable.

---

la position implicitement, grâce à la nouvelle vitesse ou vice-versa. Enfin, il peut parfois abusivement s'agir de la *méthode des trapèzes* ou d'une implicitation au premier ordre (linéarisée).

Chapitre 5

Comparaisons et résultats

Ce chapitre a pour objet la validation et la comparaison de notre modèle de Super-Clothoïdes 3D sur les plans géométrique et dynamique. Concernant la géométrie, un diagramme temps/précision permet de mettre en évidence l'efficacité de notre solution par rapport à des méthodes d'intégration d'équations différentielles. Le modèle dynamique est quant à lui comparé à deux modèles discrets de référence, eux aussi basés sur le modèle continu de Kirchhoff. Notre méthode se révèle être tout à fait compétitive dans le domaine du temps réel et est une approche à privilégier lorsque la précision est un critère important.

Si vous utilisez la version numérique de ce document, vous devriez avoir accès à la vidéo présentant nos résultats et nos comparaisons en cliquant sur l'icône dans la marge de droite, sinon, rendez-vous à l'adresse `http://bipop.inrialpes.fr/~casati/research.html#ssc`.

5.1 Calcul de la géométrie : séries entières *versus* schémas d'intégration

Une manière classique d'attaquer le problème de Darboux $4.1.13_{33}$ est de l'intégrer numériquement. Pour cela, il est préférable de représenter la base matérielle par un quaternion car la dimension du système à intégrer sera plus petite. On aura besoin d'un vecteur de $\mathbb{R}^7$, 4 composantes pour le quaternion, 3 pour la courbe moyenne. Pour comprendre pourquoi il est en revanche recommandé de conserver une représentation matricielle pour nos séries entières, on pourra se reporter à la remarque $4.1.14_{33}$. Dans ce formalisme, les équations de la base et de la courbe moyenne sont données par $4.1.15_{34}$ et $4.1.16_{34}$. Il n'est pas nécessaire d'utiliser un schéma particulièrement stable puisque l'équation n'est pas particulièrement « raide ». En revanche, il est nécessaire de maintenir le quaternion normé car seuls les quaternions de la sphère unité de $\mathbb{R}^4$, $\mathcal{S}^3(\mathbb{R})$, représentent des rotations de $\mathbb{R}^3$. Pour cela, soit on re-normalise le quaternion après chaque itération, soit on

utilise des méthodes qui garantissent que l'on reste sur une certaine variété (ici $\mathcal{S}^3(\mathbb{R})$), on parle alors d'*intégration géométrique*. Ici notre variété est un *groupe de Lie* et il existe des méthodes simples pour intégrer sur ces variétés (Hairer *et al.*, 2006).

Nous avons comparé notre algorithme de résolution du problème de Darboux aux différents schémas d'intégration suivants. Le schéma d'Euler explicite — d'ordre 1, ci-dessous dénommé « Euler » (Hairer *et al.*, 2006, équation 1.5 p. 3) —, la méthode de Lie-Euler (Hairer *et al.*, 2006, équation 8.10 p. 143) — d'ordre 1, ci-après dénommée « Lie », les schémas de Runge-Kutta d'ordre 2 et 4 — dénommés ci-dessous respectivement « RK2 » (Hairer *et al.*, 2006, système 1.3 (à droite) p. 28) et « RK4 » (Hairer *et al.*, 2006, tableau de Butcher 1.8 (à gauche) p. 30) et enfin l'algorithme (D)LSODAR de la suite ODEPACK (Hindmarsh, 1983), réputé pour sa robustesse et sa précision. Excepté pour les méthodes Lie et LSODAR, tous les schémas sont terminés par une étape de normalisation du quaternion.

Notre algorithme de comparaison est le suivant. Le problème considéré a pour paramètres $L = 0{,}2$, la torsion varie linéairement de 100 à 0, et les courbures sont égales et varient également linéairement mais de 0 à 100. On génère alors une approximation de la solution a très grande précision (avec notre méthode car c'est celle qui est la plus précise), ce sera notre solution de « référence » par rapport à laquelle seront calculées les erreurs globale en norme L_2 pour un échantillonnage uniforme de 100 points du segment $[0\,;L]$. Toutes les méthodes citées précédemment requièrent un pas d'intégration dont on suppose d'une part que l'erreur globale est une fonction strictement croissante et d'autre part que le temps total de calcul (pour intégrer de 0 à L) est une fonction décroissante. Étant donnée une erreur globale ε fixée, on peut donc trouver, par dichotomie par exemple, le plus grand pas d'intégration Δs tel que l'erreur globale de la méthode soit en dessous de ε. À ce pas Δs correspond un temps de calcul $T = \phi(\varepsilon)$. Pour chacune des méthodes, nous avons construit cette fonction ϕ qui donne le temps minimum mis par la méthode considérée pour atteindre l'erreur globale donnée. Chacun des graphes correspondants se trouve figure $5.1.1_{71}$.

Pour pouvoir comparer notre méthode sur cette même figure, il nous faut construire l'analogue de la fonction ϕ. Ce qui joue le rôle du pas d'intégration dans le calcul par séries entières, c'est l'indice de troncature de la somme. C'est en tout cas la quantité qui détermine le coût de l'algorithme. Étant donnée une erreur globale ε, on calcule alors l'indice n minimal à partir duquel on peut tronquer la somme et obtenir une erreur globale inférieure à ε. Cette troncature correspond à un temps de calcul $T = \phi(\varepsilon)$ et on peut comparer notre méthode aux autres. Elle est désignée par le sigle « DSE » sur la figure $5.1.1_{71}$. Remarquons que contrairement aux autres techniques, l'approche par séries entières ne nécessite pas d'intégrer sur tout le segment pour obtenir la valeur de la géométrie en $s = L$. Dans beaucoup d'applications, il est nécessaire de calculer un échantillonnage de la courbe et pas seulement la géométrie en $s = L$. Cet échantillonage est « gratuit » pour les méthodes d'intégration. Pour ne pas désavantager ces dernières, le temps T pour la méthode par développement en séries entières tient compte du calcul de la géométrie en 50 points répartis uniformément sur le segment $[0\,;L]$.

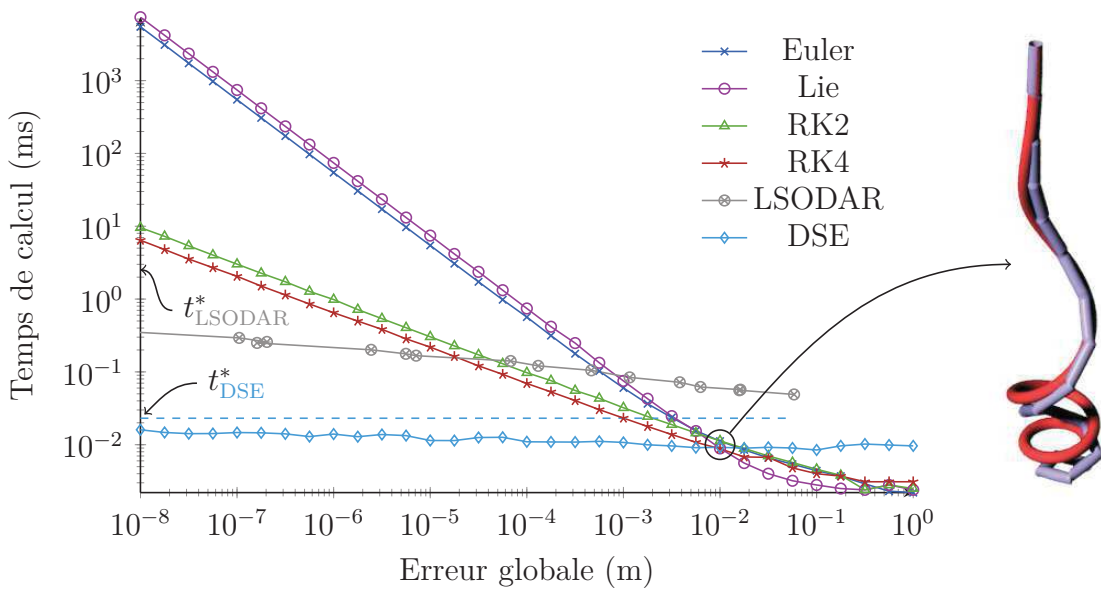


FIGURE 5.1.1 : Comparaison de notre méthode par séries entières (DSE) et plusieurs méthodes d'intégration pour la résolution du problème de Darboux dans le cas d'un vecteur de courbures affine. t^*_{DSE} est le temps mis par notre méthode pour que l'erreur globale atteigne la précision machine (en double), idem pour t^*_{LSODAR} avec LSODAR.

Interprétons ces résultats. L'ordre 1 des schémas d'Euler et de Lie est clair et l'intégration géométrique n'a pas d'intérêt dans ce cas. L'apport d'un schéma d'ordre 2 comme RK2 est bien visible mais la différence avec la méthode RK4 est faible. Ceci s'explique par la complexité plus élevée de la méthode mais surtout à cause de la normalisation du quaternion qui a pour effet de « tuer » l'ordre du schéma. LSODAR n'est pas très compétitif pour des erreurs plus grandes que 10^{-5} mais devient performant pour des erreurs plus faibles. La méthode possède en effet un « coût fixe » important[1]. L'avantage des séries entières sur les autres approches pour résoudre le problème de Darboux dans le cas d'une courbure affine est ici très net, elles sont 100 fois plus rapides que LSODAR pour atteindre la précision machine. Ceci n'a rien de mystérieux, notre méthode est spécifique et exploite beaucoup la structure du problème, ce que les approches d'intégration ne peuvent pas faire puisqu'elles sont utilisables avec la plupart des équations différentielles, même non-linéaires.

Il est intéressant de constater qu'en un certain point ($\varepsilon \approx 10^{-2}$), les graphes se croisent et toutes les méthodes (sauf LSODAR) deviennent plus précises (pour un temps fixé) que le calcul par séries entières. Pour nous rendre compte de l'erreur globale à

[1]LSODAR est capable d'intégrer des problèmes très mal conditionnés en détectant automatiquement les phases raides et les phases plus simples. Il est donc normal que cette implémentation ne soit pas partout compétitive par exemple avec un schéma RK4 implémenté spécialement pour le problème considéré. Notons que sur cet exemple, une régression log-linéaire donne un ordre 9 pour l'intégrateur LSODAR.

laquelle correspond ce point, nous avons superposé la solution exacte (en rouge) et la solution obtenue pour cette erreur globale. La méthode de calcul utilisée ici est la méthode de Lie-Euler mais ceci n'a pas beaucoup d'importance, les autres méthodes auraient donné des écarts similaires.

Notons aussi qu'à aucun moment dans nos calculs numériques par séries entières nous n'avons prêté attention au fait que la somme de la série de la base matérielle soit une matrice de rotation. Cependant, nous avons constaté numériquement que l'erreur $\left\|\mathcal{R}^\top\mathcal{R} - I\right\|$ était de l'ordre de la précision machine.

On a vu que la méthode par séries entières se révèle être un excellent moyen de calculer la géométrie d'une Clothoïde 3D à la précision machine. Son second avantage, qui n'est pas des moindres pour l'application qui est la nôtre, est que la méthode s'applique (presque) directement au calcul de la dynamique. Si l'on avait voulu calculer les termes de la dynamique avec des intégrateurs numériques, il aurait fallu résoudre les équations différentielles sur Dr, D^2r, $D\mathcal{R}$ et sur $D^2\mathcal{R}$, pour ensuite utiliser une méthode de quadrature afin d'évaluer les intégrales $4.3.11\mathrm{a}_{61}$ et $4.3.11\mathrm{b}_{61}$. Ceci se serait révélé coûteux, peut-être instable et la superposition de l'intégration et de la quadrature aurait rendu le contrôle de l'erreur peu aisé. Avec nos séries entières, quadrature et intégration sont deux opérations très simples. Notons que les coûts les plus importants ne sont pas situés aux mêmes endroits selon la stratégie de calcul utilisée : le calcul par séries entières est ralenti par les produits de Cauchy mais pas par l'intégration, alors que c'est l'inverse pour une méthode du type intégration + quadrature.

Avant de présenter notre comparaison dynamique, nous évaluons l'efficacité de notre méthode dans le cas particulier de la dimension 2, pour lequel le problème de Darboux se simplifie.

Le cas de la dimension 2

Nous l'avons déjà vu, le cas de la dimension 2 — qui consiste à ne considérer qu'une seule courbure, notons-la κ — apporte son lot de simplifications. On ne dispose toujours pas d'expression en forme fermée (voir l'annexe A_{157}) dans le cas général pour la courbe moyenne mais le repère a en revanche le bon goût d'en posséder une. L'implication principale de ceci est que le problème de Darboux, qui se formule maintenant uniquement sur la courbe moyenne, peut se mettre sous la *forme intégrale*

$$r(s) = r_0 + \int_0^s (\cos\theta(u),\ \sin\theta(u))\,\mathrm{d}u \quad \text{avec} \quad \theta(s) = \theta_0 + \int_0^s \kappa(u)\,\mathrm{d}u. \tag{5.1.2}$$

Cette formulation a l'avantage de permettre le calcul de la courbe moyenne par des algorithmes de quadrature. C'est d'ailleurs ce qui est principalement utilisé dans la littérature (Bertails-Descoubes, 2012; Kelly et Nagy, 2003; McNaughton *et al.*, 2011). Il est donc intéressant de comparer notre algorithme aux méthodes de quadrature dans ce cas.

La suite QUADPACK (Piessens *et al.*, 1983) propose des implémentations de référence pour des méthodes de quadrature efficaces. Notre problème ne comportant pas de singularité ni de discontinuité, les routines QUADPACK_QNG et QUADPACK_QAG sont adaptées à notre cas (Piessens *et al.*, 1983, p. 79). Ces routines prennent notamment en paramètre une erreur absolue qui ne permet pas de bien contrôler le temps de calcul. Pour deux erreurs absolues qui diffèrent d'un ordre de grandeur, on peut par exemple obtenir la même erreur absolue « réelle » et que les temps de calcul soient pourtant différents. Il n'est donc pas facile d'obtenir un graphique du type de $5.1.1_{71}$. Ce qui est sûr c'est que ces méthodes permettent une précision importante (de l'ordre de 10^{-10}) et l'on peut comparer le temps que met chaque algorithme pour atteindre sa précision maximum.

L'exemple comparatif que nous avons choisi est celui du calcul des intégrales de Fresnel $3.0.3_{27}$ en $s = 4$. Les deux routines QUADPACK_QNG et QUADPACK_QAG donnent ici les mêmes résultats et sont 7,5 fois plus lentes que notre méthode pour atteindre la précision maximale. Plus précisément, elles calculent la position de la spirale de Cornu en $s = 4$, avec une erreur[2] de l'ordre de $10^{-10} \approx 4{,}5 \cdot 10^5 \varepsilon$, 7,5 fois plus lentement que notre méthode ne met de temps pour atteindre une précision inférieure à 2ε. Notons que dans cet exemple, pour les séries entières, nous avons utilisé le code développé pour la dimension 3 et entre la moitié et les 2/3 des opérations sont effectuées inutilement (ce sont des multiplications de zéros). On aurait donc un gain encore plus significatif avec un code spécifiquement développé pour la dimension 2.

5.2 Comparaisons avec d'autres modèles dynamiques de tiges de Kirchhoff

Deux modèles discrets de tiges de Kirchhoff ont été introduits en informatique graphique. Il s'agit des *Super-Hélices* (SH) (Bertails *et al.*, 2006) et des *Discrete Elastic Rods* (DER) (Bergou *et al.*, 2008). Dans un premier temps nous les présenterons brièvement, avant de décrire notre protocole de comparaison puis d'en donner les résultats.

5.2.1 Présentation des modèles

5.2.1.1 Le modèle des Super-Hélices

Le modèle des Super-Hélices (Bertails *et al.*, 2006) est très proche de celui des Super-Clothoïdes 3D. Plus précisément ce dernier en est fortement inspiré. Ils partagent le concept de la discrétisation spatiale au niveau des courbures. La clef du modèle des Super-Hélices est l'exploitation de la remarque $4.1.22_{35}$ qui affirme que dans le cas où le vecteur de courbures est constant, il existe une forme fermée (voir l'annexe A_{157}) pour la base matérielle et pour la courbe moyenne. Les courbes ainsi décrites sont des

[2] Cette erreur est calculée grâce à l'utilisation de fonctions standards dédiées au calcul des intégrales de Fresnel.

hélices circulaires. Ces primitives sont ensuite raccordées de manière $\mathscr{C}^1$ en imposant la continuité de la base matérielle aux jonctions. Les composantes du vecteur de courbures sont des fonctions constantes par morceau.

Si p est le nombre d'éléments, le nombre de degrés de liberté est $3p$ puisqu'il faut 3 scalaires par élément pour d'écrire complètement la cinématique de la tige. Ceci est à comparer avec la quantité $3(p+1)$ pour les Super-Clothoïdes 3D. Le nombre de degrés de liberté par élément est donc le même asymptotiquement puisqu'il est de 3 pour les Super-Hélices et de $3 + \frac{3}{p}$ pour les Super-Clothoïdes 3D. On n'a donc pas vraiment « gagné » de degré de liberté par élément en utilisant des primitives de degré supérieur. Ceci s'explique simplement par le fait que nous imposons un degré de continuité plus fort que celui des Super-Hélices, ce qui impose une contrainte supplémentaire sur la courbure : sa continuité. On peut alors se demander si le modèle que nous avons développé est vraiment plus « riche » que les Super-Hélices. Il sera intéressant de prêter attention à cela dans les comparaisons.

De la même manière que les séries entières se « propagent » à la dynamique dans le cas des Super-Clothoïdes 3D, ce sont tous les termes de la dynamique des Super-Hélices qui s'expriment en forme fermée (voir l'annexe A_{157}). L'équation $4.3.10_{61}$ est donc écrite à base de fonctions et d'opérations élémentaires. Le schéma d'intégration temporelle est en revanche identique en tout point à celui que nous utilisons : les termes élastiques sont implicites, les autres sont explicites.

5.2.1.2 Le modèle des Discrete Elastic Rods

Le modèle des Discrete Elastic Rods (Bergou *et al.*, 2008) repose sur un échantillonnage spatial de la courbe moyenne accompagné d'une information angulaire aux points de la discrétisation. Cet angle quantifie la rotation de la base matérielle par rapport au repère de Bishop — ce repère a la particularité de ne pas tourner autour de la tangente — et permet ainsi une réduction du nombre de paramètres. Ceci garantit en particulier que la base reste « adaptée » à la courbe moyenne, c'est-à-dire que son premier vecteur est la tangente à la courbe. De ce point de vue, ce modèle est à la fois un modèle nodal et un modèle partiellement réduit.

Dans leur version de 2008, les Discrete Elastic Rods sont assorties d'une hypothèse quasi-statique simplificatrice qui permet finalement de faire « disparaître » les coordonnées angulaires comme degrés de liberté. Elles sont en fait calculées par un algorithme de type Newton–Raphson à partir de la position des points de la courbe moyenne uniquement. Dans leur version de 2010 (Bergou *et al.*, 2010), que nous utilisons dans la suite, cette hypothèse est relâchée et les angles sont conservés comme de « vrais » degrés de liberté. Si p est le nombre de segments, ce modèle dispose de $4p + 3$ degrés de liberté : 1 angle par segment et 3 coordonnées par point.

À cause de la paramétrisation nodale, l'inextensibilité n'est pas garantie et doit être traitée explicitement : soit par un algorithme de projection (Bergou *et al.*, 2008), soit par l'ajout d'une énergie d'élongation/compression assortie d'une raideur importante

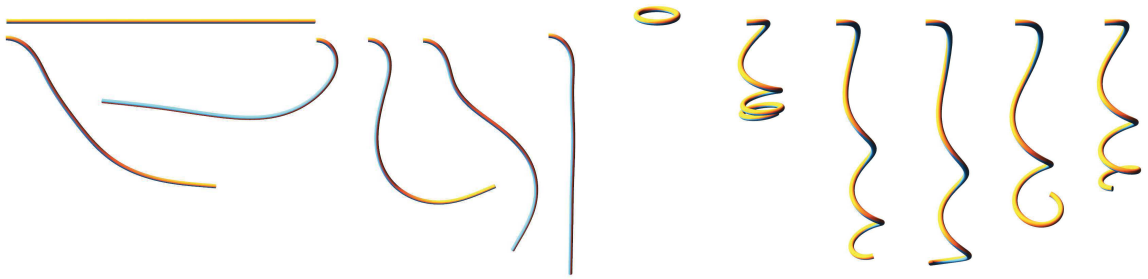
(Bergou *et al.*, 2010). Pour garantir la stabilité du modèle, un schéma implicite est utilisé et nécessite la résolution d'un algorithme de type Newton–Raphson. En revanche, la paramétrisation nodale permet d'aboutir à des systèmes creux qui peuvent être résolus efficacement, ce qui autorise l'utilisation d'un nombre important de degrés de liberté.

Précision sur les codes utilisés

Que ce soit pour la simulation du modèle des Super-Hélices ou celle des Discrete Elastic Rods, nous utilisons la version du code des auteurs, qu'ils soient d'ailleurs ici remerciés de nous l'avoir fourni. Signalons aussi que notre code de simulation est disponible en libre accès et sous licence libre (GPL) en cliquant sur l'icône dans la marge de droite ou depuis la page du projet `http://bipop.inrialpes.fr/~casati/research.html#ssc`.

5.2.2 Description du protocole de comparaison

Afin de comparer les modèles, nous utilisons deux scénarios. Le premier est celui d'une tige droite, dont une extrémité est libre et l'autre est encastrée, et qui tombe sous la gravité. Quelques images sont visibles figure 5.2.1(a). Le second décrit une tige enroulée sur elle-même (en anneau) qui tombe sous la gravité, voir la figure 5.2.1(b). Pour chacun des scénarios, la forme au repos est la forme initiale (première image à gauche, pour chaque séquence). Les comparaisons mises en œuvre dans la section suivante sont données



(a) (b)

FIGURE 5.2.1 : Les deux scénarios de comparaison que nous utilisons. La flèche du temps va de gauche à droite. Les positions les plus à droite sont les positions d'équilibre.

ci-dessous.

(i) **Validation croisée.** Une première chose à vérifier est que les trois modèles convergent vers la même position d'équilibre. On effectue ainsi une validation croisée.

(ii) **Précision statique.** Pour un nombre de degrés de liberté donné, quel est le modèle qui permet d'obtenir la plus grande précision sur l'état d'équilibre ? Cette comparaison a l'avantage d'être indépendante du schéma d'intégration en temps.

En pratique, l'erreur sera calculée par notre méthode avec 50 éléments (153 degrés de liberté) car c'est celle qui converge le plus vite vers la géométrie continue.

(iii) **Efficacité des calculs dynamiques.** On fixe le pas d'intégration à 11ms et on simule une seconde de chaque scénario. Les mouvements étant assez nerveux, la tige a le temps de subir une phase de descente puis de remontée. Étant donné un nombre de degrés de liberté, quel est le modèle qui est le plus rapide pour effectuer ce calcul ?

(iv) **Pseudo-diagramme temps/précision.** Comparer les mouvements dynamiques en mesurant l'erreur à chaque pas ne nous apporterait pas beaucoup d'information. En effet, une telle comparaison repose essentiellement sur la précision du schéma d'intégration utilisé. Chaque modèle pouvant être couplé avec à peu près tous les schémas d'intégration, cette comparaison n'est pas très pertinente. D'autre part, ce sont ici le choix de la discrétisation géométrique (spatiale) et son impact sur le coût de la simulation dynamique qui nous intéressent. Pour associer un temps de calcul à une précision, il nous a semblé approprié de « croiser » les deux comparaisons précédentes. C'est-à-dire associer un temps de calcul dynamique à une précision statique via un nombre de degrés de liberté. On obtient ainsi une estimation de la précision dynamique que l'on pourrait espérer si l'on avait un schéma d'intégration parfait (sans prendre en compte toutefois le coût d'un tel schéma).

5.2.3 Comparaisons

(i) **Validation croisée.** Les formes à l'équilibre prédites par les trois modèles convergent vers la même courbe moyenne et la même base matérielle lorsque le nombre de degrés de liberté tend vers l'infini. Ceci est déjà un résultat en soi car la validation croisée des modèles des Discrete Elastic Rods et des Super-Hélices n'avait jamais été vérifiée.

(ii) **Précision statique.** Le résultat de cette expérience est présenté figure $5.2.2_{77}$. Précisons que l'erreur est mesurée en norme L_2 pour un échantillonnage uniforme de 100 points quel que soit le nombre de degrés de liberté. Afin de « masquer » le faible degré de continuité des DER, celles-ci sont souvent lissées avec des courbes paramétriques de type Spline. Cependant, ceci n'a pas pour effet de faire diminuer l'erreur sur le graphe de la figure $5.2.2_{77}$ et nous ne l'utilisons donc pas.

L'ordre de convergence de chacune des méthodes est bien visible ici et l'apport des Super-Clothoïdes 3D sur les autres modèles est clair.

Par exemple, il faut 2,7 fois moins de degrés de liberté à notre modèle pour atteindre la limite de distinction visuelle avec la référence qu'il n'en faut au modèle des Super-Hélices. Il lui en faut 40 fois moins qu'aux Discrete Elastic Rods.

(iii) **Efficacité des calculs dynamiques.** Les temps de simulation en fonction du nombre de degrés de liberté sont donnés figure $5.2.3_{78}$. Le caractère creux du sys-

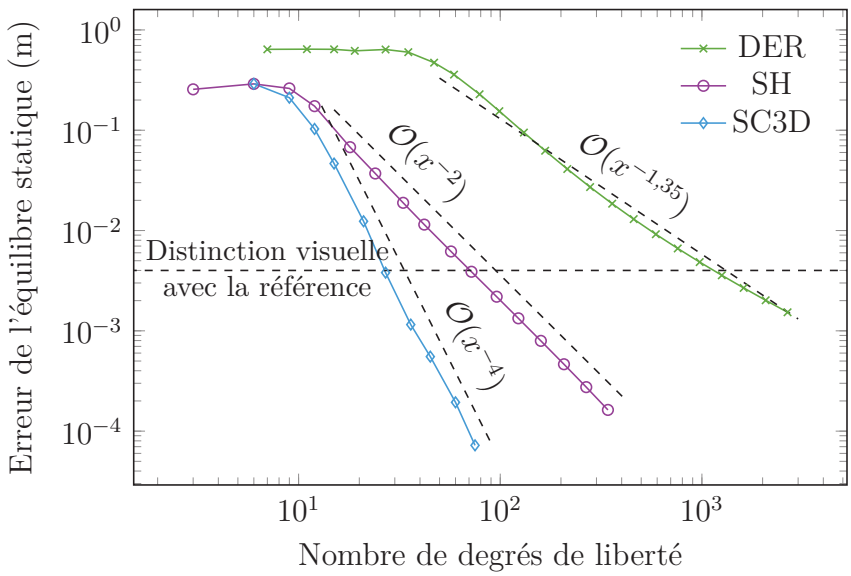


FIGURE 5.2.2 : Comparaison de la précision des modèles sur l'équilibre statique en fonction du nombre de degrés de liberté.

tème des Discrete Elastic Rods lui confère l'efficacité attendue. Le coût des calculs par séries entières est quant à lui mis en évidence par comparaison avec les Super-Hélices. Rappelons en effet que pour un nombre de degrés de liberté n donné, le nombre d'éléments de ces deux modèles est quasiment identique puisqu'il est égal à $\frac{n}{3} - 1$ dans le premier cas et $\frac{n}{3}$ dans le second. La taille des systèmes linéaires à inverser, pour une abscisse fixée sur le graphique, est donc pratiquement la même.

On remarque que le modèle des Discrete Elastic Rods fonctionne en temps réel pour cette simulation jusqu'à une vingtaine d'éléments ; 18 pour les Super-Hélices et 10 pour les Super-Clothoïdes 3D.

(iv) **Pseudo-diagramme temps/précision.** Le « croisement » des deux graphiques précédents, en éliminant le nombre de degrés de liberté afin de conserver la précision en fonction du temps de calcul, est donné figure 5.2.4$_{79}$. Rappelons que ce graphique est donné à titre indicatif et ne constitue pas une évaluation de la précision dynamique des modèles. C'est, comme il est dit plus haut, une estimation de la précision dynamique que l'on pourrait espérer si l'on avait un schéma d'intégration parfait.

Il est intéressant de constater que les graphes des Super-Hélices et des Super-Clothoïdes 3D se croisent et que le point d'intersection se trouve au-dessus de la limite de distinction visuelle et avant la limite du temps réel. Dans la région de faible précision (erreur $> 2 \cdot 10^{-2}$), les Super-Hélices offrent un meilleur compromis que notre modèle. Cependant, pour de telles précisions, le nombre d'éléments des Super-Hélices est inférieur à 6 et laisse apparaître les discontinuités peu esthétiques dont il est question dans les motivations (voir la figure 3.0.2$_{26}$). Pour des précisions

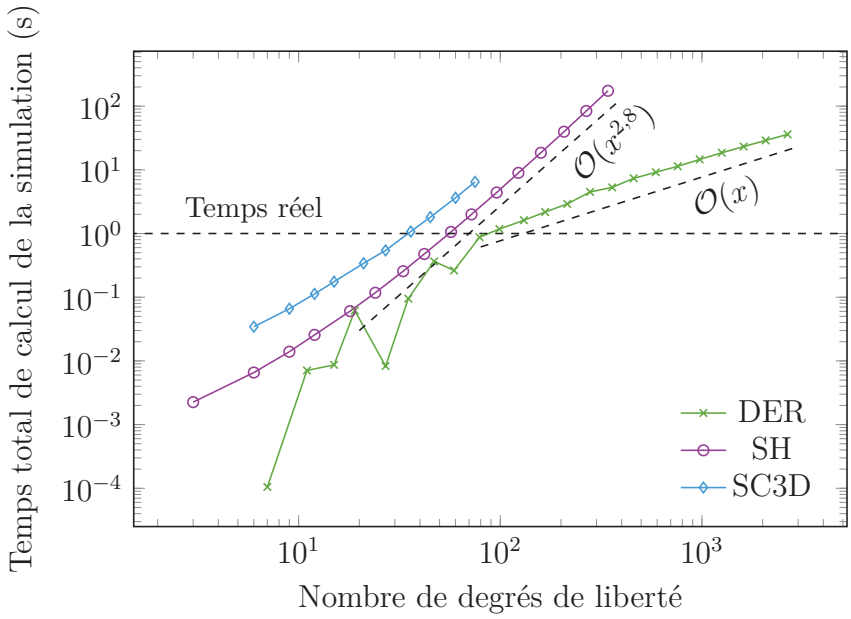


FIGURE 5.2.3 : Temps de calcul de la simulation en fonction du nombre de degrés de liberté pour chaque modèle.

plus importantes, notre modèle devient plus intéressant que les Super-Hélices et le surcoût de calcul est compensé par le gain de précision.

Pour des mouvements « nerveux » comme ceux que nous avons considérés, les Discrete Elastic Rods sont peu compétitives par rapport aux deux autres modèles. Elles sont particulièrement pénalisées dans cette comparaison par le fait que la géométrie fortement courbée requiert un nombre important de points pour être correctement décrite par des segments. De plus, la rapidité et la brutalité du mouvement mettent le schéma numérique à l'épreuve et l'algorithme de Newton–Raphson requiert beaucoup d'itérations pour converger. En traitant l'inextensibilité de manière implicite et en implicitant les termes de flexion/torsion dans le schéma, les deux autres modèles s'en sortent beaucoup mieux. Le fait que certains termes soient explicites dans notre schéma a aussi pour conséquence de permettre à notre modèle de moins dissiper l'énergie comme le montre la figure 5.2.5$_{80}$.

Discussion Soulignons le fait que notre étude est modeste et limitée, mais une comparaison exhaustive serait de toute façon difficile. Chaque modèle a ses forces et ses faiblesses et le meilleur compromis est à définir en fonction du contexte d'utilisation. Cependant, notre étude a montré que dans le cas de mouvements nerveux et de courbures importantes, le modèle des Super-Clothoïdes 3D est le meilleur compromis en termes de rapidité et de précision. La force principale des Discrete Elastic Rods est définitivement sa grande stabilité à un coût linéaire (figure 5.2.3). En revanche, l'inconvénient d'une

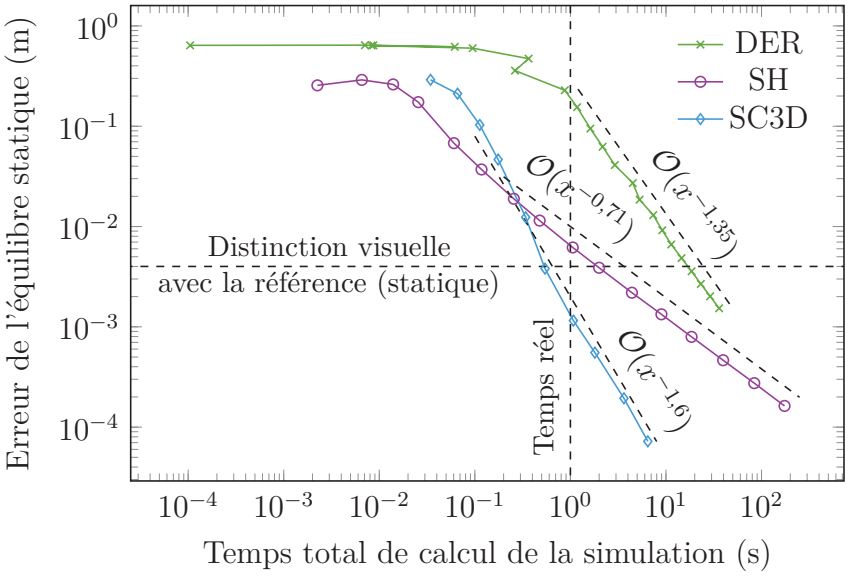


FIGURE 5.2.4 : Croisement des graphiques $5.2.2_{77}$ et $5.2.3_{78}$ par « élimination » du nombre de degrés de liberté. Remarquons qu'une fonction en x^a sur le graphique $5.2.2_{77}$ croisée avec une fonction en x^b sur le graphique $5.2.3_{78}$ est en $x^{\frac{a}{b}}$ ici.

telle approche est le coût important de convergence de la boucle de Newton–Raphson dans les parties numériquement « raides ». Rappelons que chaque modèle pourrait être utilisé avec un schéma numérique plus performant, du point de vue de la stabilité, de la précision ou de la conservation de l'énergie (Hairer *et al.*, 2006). Dans notre étude, nous avons utilisé les modèles avec le schéma d'intégration proposé dans la version originale du code des auteurs.

5.3 Résultats

Notre modèle a été utilisé pour reproduire des phénomènes mécaniques complexes mettant en jeu des tiges fortement courbées, comme la simulation de plantes grimpantes volubiles (qui s'enroulent autour d'un support), ou celle de cheveux bouclés. Nous présentons ces résultats ici.

5.3.1 Plantes grimpantes volubiles

La croissance des plantes grimpantes volubiles est un phénomène tout à fait singulier qui soulève beaucoup de questions. Quelle taille doit avoir le support pour que la plante puisse effectivement s'y accrocher et grimper ? Quelle est la topologie de contact entre la tige et le support ? Goriely et Neukirch (2006) ont modélisé ce phénomène comme un problème aux limites grâce à une tige de Kirchoff bidimentionnelle de courbure au repos $\bar{\kappa}$, constante, en contact (sans frottement) à ses deux extrémités avec un disque de rayon $R > \bar{\kappa}^{-1}$ (sinon, il est clair que la tige ne peut pas s'enrouler autour du support).

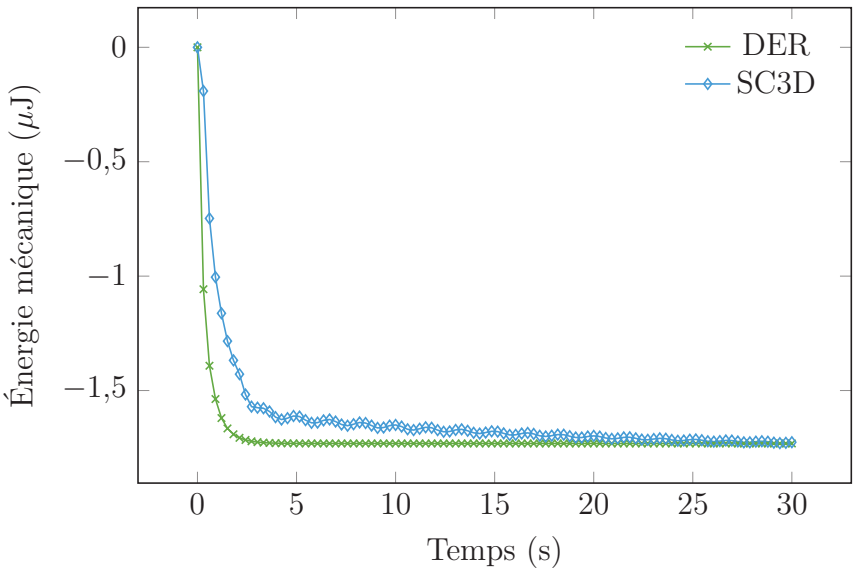


FIGURE 5.2.5 : Comparaison de la dissipation d'énergie due au schéma (il n'y a pas de dissipation physique) pour le mouvement $5.2.1(a)_{75}$.

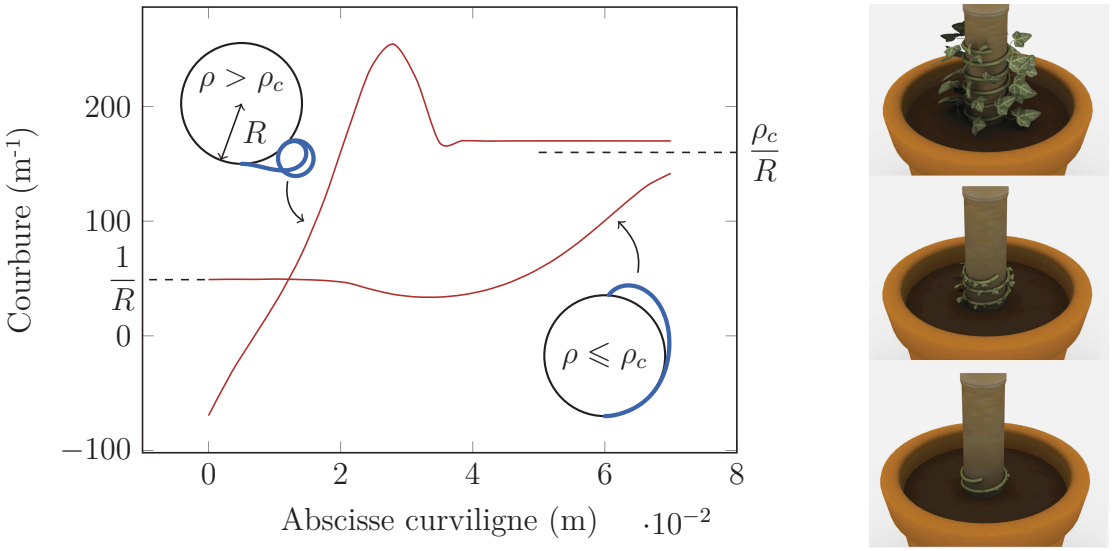
En étudiant les bifurcations du système, ils ont montré l'existence d'une valeur critique ρ_c du produit $\rho = R\bar{\kappa}$ en dessous de laquelle la plante est capable de grimper en s'enroulant autour du support. Une méthode de tir[3] leur permet de donner une valeur approchée de $\rho_c \approx 3{,}3$. Pour des paramètres favorables à l'ascension ($\rho \leqslant \rho_c$), la topologie de contact reste la même au cours du temps et est caractérisée par un point de contact à l'extrémité (qui fait un angle constant avec le support), suivi d'une région qui n'est pas en contact et enfin d'une zone continue de contact entre le support et le reste de la plante. Dans le cas défavorable ($\rho > \rho_c$), la plante s'enroule sur elle-même sans s'accrocher au support.

Nous avons utilisé notre propre modèle numérique pour simuler dynamiquement le processus de croissance en modélisant les forces de contact avec des pénalités. La figure $5.3.1(a)_{81}$ montre que nous avons capturé convenablement tous les phénomènes. Nous avons retrouvé la valeur critique ρ_c avec une erreur relative de 1% avec seulement 5 éléments de Clothoïdes 3D (utilisées en 2D ici).

Dans le cas tridimentionnel, les auteurs prédisent une forme hélicoïdale dont le pas n'est pas uniforme. Nos simulations confirment ceci surtout près de l'extrémité, voir la figure $5.3.1(b)_{81}$.

Notons que nous n'avons utilisé que 10 éléments de Clothoïdes 3D qui nous ont permis de faire plus de 5 tours autour du support.

[3]Les méthodes de tir s'appliquent à la résolution des problèmes aux limites où les conditions (inconnues) à un bord sont approchées itérativement pour satisfaire les conditions au bord opposé, par des résolutions successives du problème de Cauchy correspondant.



(a) (b)

FIGURE 5.3.1 : Simulation d'une plante grimpante volubile. En 2D (a), on retrouve le seuil critique $\rho_c \approx 3{,}3$ et la topologie de contact prédits par Goriely et Neukirch (2006). En 3D (b), l'utilisation de 10 éléments nous permet de simuler convenablement la croissance de la plante. Observer la variation du pas de l'hélice près de l'extrémité.

5.3.2 Simulation de chevelure

Notre modèle a été facilement couplé avec le solveur de contact frottant de Daviet *et al.* (2011). Nous l'avons utilisé pour simuler différents scénarios mettant en jeu du contact cheveux/cheveux et cheveux/personnage.

Dans un premier temps, nous avons animé deux modèles constitués tous les deux de 662 tiges (cheveux) avec 4 éléments de Clothoïdes 3D chacun. Les courbures au repos du premier modèle sont droites, celles du second sont assez fortement courbées. Des captures de ces simulations sont montrées figure 5.3.2$_{82}$.

Nous avons aussi animé une chevelure fantaisiste inspirée de « Animation Mother » de Siggraph (CGSociety, 2008). La chevelure est constituée de 100 tiges en contact frottant composées de 1 à 4 éléments de Clothoïdes 3D (soit 1,55 élément en moyenne seulement). Le résultat est donné figure 5.3.3$_{82}$. Les spirales ont un mouvement riche même avec peu d'éléments et l'emmêlement complexe est bien géré par le solveur de contact.

5.3.3 Animation de rubans

Enfin, même si les hypothèses des tiges minces ne sont plus satisfaites dans le cas de rubans (Dias et Audoly, 2015), nous avons modélisé un ruban en utilisant une tige de Kirchhoff avec une section exagérément plate. Un seul élément de Clothoïde 3D permet

(a) (b)

FIGURE 5.3.2 : Simulation de 662 cheveux droits (a) et frisés (b) en contact frottant entre eux et avec le personnage. Le coefficient de frottement vaut $\mu = 0{,}2$.

FIGURE 5.3.3 : Simulation fantaisiste de « Animation Mother », de Siggraph (CGSociety, 2008).

déjà des mouvements riches et des formes complexes tout en étant très efficace. Nous avons utilisé cette stratégie pour simuler en temps réel la géométrie d'un ruban enroulé, d'un parchemin et d'un poster comme le montre la figure 5.3.4$_{83}$.

Ces expériences numériques illustrent la richesse géométrique de notre primitive ainsi que la variété de ses déformations mécaniques.

(a) (b) (c)

FIGURE 5.3.4 : Simulation d'un ruban enroulé (a), d'un parchemin (b) et d'un poster (c), chacun possédant un seul élément de Clothoïde 3D. Les hypothèses de validité du modèle de Kirchhoff ne sont plus satisfaites mais les formes ainsi générées sont tout de même très riches et visuellement plausibles, pour un coût de simulation très faible.

Chapitre 6

Conclusion de la première partie

Résumé des contributions Dans cette première partie, nous avons proposé un modèle discret de tiges de Kirchhoff dynamiques, de haut degré, basé sur des éléments en courbures et torsion affines par morceaux : les Super-Clothoïdes 3D. Cette discrétisation spatiale est calculée de manière précise grâce à une méthode dédiée, adaptée à l'arithmétique flottante, utilisant des développements en séries entières. L'approche s'est révélée être beaucoup plus efficace et précise que les intégrateurs numériques classiques. D'autre part, nous avons montré que l'algorithme utilisé pour la résolution du problème géométrique s'appliquait aisément au calcul de tous les termes de la dynamique. Enfin, comme dans (Bertails *et al.*, 2006), l'utilisation des courbures et de la torsion comme degrés de liberté a permis d'aboutir à un schéma d'intégration stable grâce à une implicitation, à moindres frais, des forces élastiques. Le modèle a été utilisé avec succès pour simuler des tiges, droites ou courbées, comme des cheveux, des plantes ou des rubans ; dans tous les cas, un petit nombre d'éléments s'est avéré suffisant. Nos comparaisons avec deux modèles de référence introduits en informatique graphique ont montré que notre approche offrait un meilleur compromis en termes de précision spatiale, de richesse de mouvements et d'efficacité en temps de calcul.

Travaux futurs Une extension évidente et dont nous avons déjà discutée dans la section $4.2.4_{55}$ est celle de l'utilisation d'éléments de courbures et torsion polynomiales de degrés quelconques, raccordés avec un degré de continuité, lui aussi quelconque. Une telle flexibilité permettrait de pouvoir choisir entre beaucoup d'éléments de degré faible et peu d'éléments de degré élevé. Le cas d'un seul élément de degré élevé semble même intéressant lorsque l'on souhaite une précision importante. De la figure $5.2.2_{77}$, on peut conjecturer que cette précision (sur la position statique) est en $\mathcal{O}\big(x^{2d}\big)$ avec le nombre de degrés de liberté si d est le degré du polynôme utilisé sur chaque élément. Remarquons aussi que le temps de calcul est linéaire en d puisque celui-ci n'influe que sur le nombre de termes de la somme intervenant dans la récurrence $4.2.4\mathrm{a}_{41}$. Tout ceci semble confirmer la pertinence de l'élaboration d'un tel modèle. Une autre amélioration possible de la méthode, dont nous avons déjà discutée à la section $4.3.3_{64}$, est celle de l'élimination des produits de Cauchy qui sont responsables de 72% du temps de calcul (en moyenne). Une

piste intéressante serait de trouver les équations différentielles satisfaites par ces produits de manière à obtenir une relation de récurrence sur leurs coefficients, et de pouvoir ainsi traiter ces séries comme les autres. Aussi, l'utilisation d'un autre type de séries pourrait présenter plusieurs avantages. En effet, il est possible que pour un autre type de séries, le phénomène de compensation catastrophique soit absent. Ou peut-être que celui-ci est moins problématique et que de telles séries permettraient de plus grands pas de découpe. D'autre part, la majoration explicite $4.2.5_{41}$ du théorème $4.2.1_{40}$ ainsi que la condition $C(s) \leqslant T$ pourraient peut-être permettre d'obtenir des bornes d'erreurs (explicites) qui tiennent compte de toutes les erreurs d'arrondi. En effet, dans notre exposition, il a peut-être semblé que découper plus était toujours bénéfique pour la précision, mais plus de segments signifie aussi plus d'erreurs propagées et donc sûrement une erreur globale plus importante. Pour la dynamique, l'utilisation d'un schéma mieux adapté à la préservation de l'énergie serait avantageuse, mais ce genre de schémas nécessite, la plupart du temps, la résolution d'une équation implicite. Cela limite l'intérêt de la dépendance linéaire des forces par rapport aux courbures qui nous avait facilité leur implicitation. Il est aussi sûrement possible d'améliorer notre comparaison des modèles, en considérant d'autres scénarios, de manière à bien cerner le domaine le plus adapté à chacune des approches.

Deuxième partie

Modélisation numérique inverse de coques élastiques en contact frottant

Chapitre 7

Introduction

Selon Kern (2002), un *problème inverse* consiste à déterminer les *causes* connaissant les *effets*, à l'inverse du *problème direct* qui s'attache à donner les effets en fonction des causes. Pour expliciter ceci, prenons l'exemple d'une masse attachée à un ressort qui pend sous la gravité. Notons k la constante de raideur du ressort, ℓ_0 sa longueur à vide, m la masse et g la pesanteur, comme sur la figure 7.0.1.

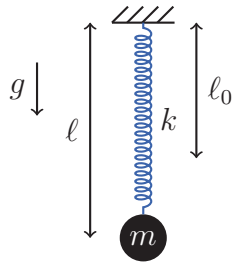


Figure 7.0.1 : Une masse attachée à un ressort qui pend sous la gravité.

L'équation de l'équilibre du système s'écrit

$$mg = k(\ell - \ell_0). \tag{7.0.2}$$

Elle donne la solution du problème *direct* puisque si l'on connaît la masse, la longueur à vide et la raideur (les *causes*), on peut déterminer la longueur ℓ du ressort à l'équilibre (les *effets*). Si en revanche on observait un tel ressort sous la gravité, connaissant le rapport $\frac{m}{k}$ et mesurant la longueur ℓ, on pourrait en déduire la longueur à vide ℓ_0 grâce à l'équation (inverse)

$$\ell_0 = \ell - \frac{m}{k}\, g. \tag{7.0.3}$$

Ceci amène deux remarques. D'abord, on a supposé qu'une partie des paramètres était connue, en effet, nous disposions d'une seule mesure pour trois paramètres à estimer ℓ_0,

k et m. On aurait aussi pu calculer la raideur (sur la masse) en fonction de la longueur au repos mais dans tous les cas il faut supposer qu'une partie des paramètres est connue ou estimée par un autre moyen. Le problème opposé, où l'on dispose de plus d'observations que de paramètres, est aussi souvent rencontré. Dans cet exemple simple, il semble que les deux problèmes soient identiques, ce sont en fait deux classes de problèmes inverses bien distincts en génie mécanique. Lorsque les paramètres physiques sont connus et que l'on cherche à estimer les paramètres géométriques, on parle de *problème de conception inverse*, dans le cas inverse, on parle de *problème de mesure inverse*. Dans la suite nous nous intéresserons à la première catégorie car nous supposerons que les paramètres physiques sont connus et fournis par l'utilisateur. La seconde remarque concerne la contrainte qui existe sur ℓ_0 : cette quantité doit être positive. Ceci n'est pas garanti par l'équation $7.0.3_{89}$ et une valeur aberrante de $\frac{m}{k}$ peut conduire à une longueur à vide négative, ce qui n'a pas de sens physique.

Symboliquement, si on note ϕ le problème direct dépendant des paramètres contenus dans la variable a, et u l'état du système associé (les effets), on peut simplement noter $\phi(a) = u$. De manière formelle, le problème inverse pourrait s'écrire $a = \phi^{-1}(u)$. Ceci est schématisé figure 7.0.4.

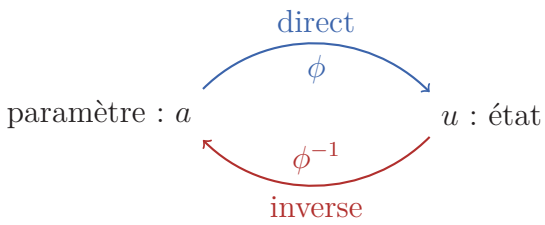


FIGURE 7.0.4 : Une vue schématique du problème direct et du problème inverse.

Beaucoup de problèmes se formulent comme des problèmes inverses. En effet, dès que l'on possède des observations d'un phénomène (des mesures) ainsi qu'un modèle de celui-ci et que l'on souhaite en tirer des prédictions, il est d'abord nécessaire de trouver les paramètres du modèle qui pourraient « expliquer » ces observations pour ensuite en déduire des prédictions.

Dans notre cas, ce qui joue le rôle des « observations », ce sont les modèles géométriques, issus de la capture ou élaborés par des infographistes, et qui sont destinés à être animés. Ces modèles sont capturés/conçus dans une configuration physique d'équilibre statique, le plus souvent sous la gravité et en contact avec d'autres objets. Malheureusement, dans la plupart des cas, on ne dispose pas des paramètres qui permettraient au simulateur physique non seulement d'interpréter effectivement la forme comme un équilibre statique mais aussi d'en « prédire » le mouvement. Dans cette thèse on considère que les observations sont totales et exactes : on connaît complètement et exactement la géométrie de l'objet à inverser ainsi que celle des objets avec lesquels il est en contact[1].

[1]En pratique, nous verrons que ceci n'est pas toujours vérifié, notamment avec les modèles issus de

Un exemple de succès récent de la formulation de ce problème en un problème inverse, dans le cas des cheveux, a été celui de Derouet-Jourdan *et al.* (2013a). Dans ce travail, une chevelure capturée ou modélisée par un infographiste, est interprétée comme un équilibre sous la gravité et en présence de contact frottant. Elle est ensuite simulée de manière réaliste alors que l'approche naïve, consistant à simuler directement le système, sans l'inverser[2], ne permet pas de conserver la forme de départ. La figure 7.0.5 montre entre autres que la simulation directe produit un affaissement de la forme prise en entrée.

(a) (b) (c)

FIGURE 7.0.5 : Un problème de conception inverse de chevelures, résolu par Derouet-Jourdan *et al.* (2013a). Une coiffure numérisée (capture) en (a) et sa simulation naïve, sans inverser le problème, en (b). L'affaissement est bien visible. En revanche, la méthode d'inversion de Derouet-Jourdan *et al.* (2013a), qui prend en compte les contacts des cheveux entre eux et avec le corps du personnage, permet de conserver les cheveux en place (à l'équilibre) lorsque la gravité est appliquée au démarrage de la simulation (c).

Le modèle de tiges élastiques utilisé dans ces travaux est celui des Super-Hélices, dont nous avons largement parlé dans la première partie de ce manuscrit. Le succès de la méthode tient en grande partie au fait que le modèle des Super-Hélices est « adapté » à l'inversion dans le sens où le problème (sans contact) est linéaire par rapport à la forme au repos des cheveux. La difficulté réside alors dans la prise en compte du contact frottant mais aussi dans la « conversion géométrique ». En effet, c'est une suite de points qui modélise la position des cheveux en entrée de la méthode, alors que la paramétrisation du modèle des Super-Hélices utilise les courbures. L'algorithme des *tangentes flottantes* (Derouet-Jourdan *et al.*, 2013b) est alors avantageusement utilisé pour effectuer cette conversion. Le contact est quant à lui résolu en reformulant le problème qui peut alors être envisagé avec le même solveur de contact que celui utilisé pour la dynamique (Daviet *et al.*, 2011).

Comme les Super-Hélices, le modèle de tiges que nous avons élaboré dans la première partie possède les atouts qui facilitent son inversion (une dépendance linéaire des forces

la capture qui peuvent être mal reconstruits ou à cause du fait que l'on ne possède pas la géométrie des objets en contact.

[2]Plus précisément, la géométrie à inverser est utilisée comme forme au repos du système physique.

élastiques en la forme au repos). L'utilisation d'algorithmes de conversion d'une suite de points en une clothoïde par morceaux a d'ailleurs permis d'inverser facilement le modèle 2D des Super-Clothoïdes (Bertails-Descoubes, 2012). Même s'il reste d'importants enjeux en inversion de chevelures (notamment en présence de contact frottant), cette piste de recherche a déjà été bien entamée et nous avons voulu savoir s'il était possible d'étendre ces contributions à d'autres types de systèmes. C'est pourquoi, nous nous proposons dans cette seconde partie d'aborder le problème de l'inversion de vêtements en présence de contact frottant qui est lui aussi un problème important et encore ouvert en informatique graphique.

L'inversion de vêtement pour l'informatique graphique

La simulation de vêtements en informatique graphique est en plein essor depuis quelques années et permet maintenant d'obtenir des animations très réalistes, qu'il est difficile de distinguer de la réalité. D'autre part, la variété des modèles simulés s'est accrue, allant d'un simple morceau de tissu à une robe constituée de plusieurs couches de tissus différents. Une telle diversité de formes et de mouvements encourage les artistes à créer des vêtements personnalisés qui contribuent à améliorer l'ensemble de la narration.

Cependant, la modélisation de vêtements virtuels reste une tâche pénible et difficile pour un infographiste, les outils actuels d'édition offrant peu de contrôle sur la forme finale (simulée) du vêtement. D'autre part, beaucoup d'outils perfectionnés permettent la confection de vêtements à partir de patrons comme on le ferait dans la réalité. L'utilisateur modélise ses patrons en 2D, indique les coutures et ajuste finalement le résultat itérativement grâce au simulateur physique (Volino *et al.*, 2005; Marvelous Designer, 2010; Umetani *et al.*, 2011). Même si cette méthode permet d'obtenir des patrons réalistes qui peuvent ensuite être fabriqués, un tel procédé à base d'essais et erreurs est peu intuitif lorsque l'on a une idée précise de la forme à laquelle on veut aboutir. D'autre part, plusieurs outils d'édition directe et intuitive comme la modélisation à base de croquis (Turquin *et al.*, 2007) ou la capture à partir d'images (Bradley *et al.*, 2008) permettent la confection semi-automatique de vêtements 3D depuis une référence virtuelle ou réelle. De tels outils géométriques laissent l'utilisateur se concentrer sur la forme finale du vêtement en donnant un contrôle important sur les détails géométriques. Cependant, la forme résultante correspond souvent à une configuration d'équilibre soumise à des forces extérieures mais ne peut pas être facilement interprétée physiquement. Si, naïvement, on donne une telle forme à l'équilibre au simulateur, en tant que forme au repos (non déformée), celle-ci va s'affaisser lorsque la simulation va être démarrée, gâchant ainsi tous les efforts de la modélisation[3], comme sur la figure $7.0.5_{91}$ (dans le cas des cheveux).

[3]Dans l'exemple du ressort du début de ce chapitre, ceci revient à lancer la simulation en prenant pour le paramètre ℓ_0 la longueur mesurée sous la gravité ℓ. Le système ne sera évidement pas à l'équilibre au début de la simulation et la masse va irrémédiablement tomber plus bas (c'est l'affaissement).

Dans cette thèse, nous développons une nouvelle stratégie pour la modélisation de vêtements virtuels de manière réaliste et intuitive. Notre idée est de concevoir une méthode inverse, automatique, qui interprète une forme géométrique comme l'équilibre stable d'un simulateur physique. Les propriétés du matériau, comme la masse surfacique, les raideurs et les coefficients de frottement, étant supposées connues, notre algorithme prend en entrée une surface 3D, issue d'une capture ou d'une modélisation géométrique, appelée *forme objectif*, et retrouve automatiquement une forme au repos plausible (celle-ci n'est pas nécessairement plate à cause des coutures dont la position est inconnue) ainsi que les forces de contact frottant en jeu. Une fois la simulation lancée avec de tels paramètres, l'équilibre trouvé est très proche de la forme objectif et peut ensuite être animé.

Sur la figure 7.0.6$_{94}$, nous montrons que la prise en compte du contact frottant est importante car le frottement statique permet d'expliquer beaucoup de formes qui ne pourraient tout simplement pas exister sans frottement.

Plan de la partie

Dans le chapitre suivant, nous donnons un état de l'art abordant les approches de simulation et de modélisation de vêtements en informatique graphique. Le chapitre 9$_{99}$ sera consacré au choix d'un modèle de coques élastiques. Même si l'omniprésence des modèles nodaux pour la simulation de vêtements en informatique graphique permet une grande application de la méthode, nous verrons qu'il n'est pas forcément le plus judicieux de privilégier de tels modèles pour l'inversion. Nous donnons des pistes de réflexion pour l'élaboration d'un modèle adapté à l'inversion en fin de chapitre. Au chapitre 10$_{115}$, nous détaillons l'approche que nous avons retenue pour retrouver la forme au repos d'une coque soumise à la gravité et à des forces de contact frottant. En nous appuyant sur la méthode de l'adjoint, guidée par une pénalisation adaptée, et en formulant la recherche des équilibres comme un problème de minimisation, nous montrons avec les résultats du chapitre 11$_{141}$ que notre approche permet effectivement d'envisager la résolution de tels problèmes.

(a) (b) (c) Daviet *et al.* (2011)

FIGURE 7.0.6 : Des systèmes mécaniques en équilibre statique pour lesquels le contact frottant joue un rôle important. Figure (a), un morceau de tissu est en équilibre sur une sphère, sans frottement, il est parfaitement centré sur celle-ci. Un léger décalage ferait tomber le tissu qui glisserait sur la sphère. En l'absence de frottement, le système est instable. La figure (b) représente un carré de tissu en contact avec une sphère et un plan. En haut, le coefficient de frottement $\mu = 0{,}3$ assure que le tissu est en équilibre sur la sphère. En bas, on a pris $\mu = 0{,}01$ et le tissu a glissé sur la sphère. Enfin, la figure (c) présente des chevelures en équilibre statique dans lesquelles les motifs complexes de cheveux, dus au frottement statique, sont bien visibles.

Chapitre 8

État de l'art

La modélisation et la simulation de vêtements ont déjà une histoire riche dans les domaines de l'informatique graphique et de la conception géométrique assistée par ordinateur. En moins de 30 ans, des solutions permettant la modélisation de vêtements complexes ainsi que la simulation à un niveau de détail important ont été proposées. De telles avancées ont été intégrées dans les standards de conception de l'industrie du divertissement, permettant ainsi leur utilisation dans les effets spéciaux ou les jeux vidéos. Nous discutons ici les différentes approches de simulation de vêtements avant de nous intéresser aux techniques actuelles de modélisation de vêtements réalistes. Une revue récente des différents travaux sur le sujet peut être trouvée dans (Liu *et al.*, 2010).

Simulation de vêtements

Depuis les années 90, une quantité importante de travaux sur la simulation de vêtements a été publiée en informatique graphique. Les travaux de (Baraff et Witkin, 1998) ont été les premiers à permettre la simulation stable de vêtements complets en des temps de calculs raisonnables, grâce à l'utilisation d'un schéma implicite (au premier ordre) de discrétisation des équations dynamiques. Les travaux suivants ont contribué au traitement robuste du contact et du frottement (Bridson *et al.*, 2002), à la meilleure formulation du terme de flexion (Bridson *et al.*, 2003; Grinspun *et al.*, 2003) ou encore à la limitation de l'élongation/compression (Goldenthal *et al.*, 2007; English et Bridson, 2008; Thomaszewski *et al.*, 2009). Les défis actuels concernent l'utilisation des propriétés réelles des matériaux (Wang *et al.*, 2011) ainsi que l'interactivité à haute résolution (Kim *et al.*, 2013; Hahn *et al.*, 2014).

Malgré la profusion de modèles de simulation, remarquons qu'ils partagent tous la même représentation nodale du vêtement. Les vêtements sont ainsi toujours modélisés par un maillage dont les points 3D sont les degrés de liberté du système dynamique correspondant.

Conception de patrons

Le processus classique de conception de vêtements réels repose sur la confection de patrons 2D, des pièces de tissu plates qui sont ensuite cousues ensemble pour créer le vêtement final. Les premières méthodes de confection virtuelle de vêtements imitaient ce processus (Carignan *et al.*, 1992). Pour rendre le processus d'édition plus intuitif, Volino *et al.* (2005) ont imaginé une méthode interactive où l'utilisateur peut simultanément éditer les patrons 2D et visualiser le résultat en 3D sous la gravité et en contact avec le corps du personnage grâce à une méthode de recherche d'équilibre rapide. Un tel environnement interactif de confection est maintenant un standard utilisé par des outils commerciaux comme le célèbre Marvelous Designer (2010). Ceci a aussi inspiré des travaux dans le milieu académique (Umetani *et al.*, 2011), qui ont amélioré la réactivité de la génération de la forme en 3D, ce qui a grandement contribué à améliorer la qualité de l'outil.

La conception de patrons virtuels est un outil puissant dans le sens où il permet à un expert de la confection de vêtements de créer et d'expérimenter des assemblages complexes sans avoir à les construire effectivement. De plus, le résultat final peut non seulement être utilisé pour fabriquer le vêtement (réel) mais aussi habiller des personnages virtuels de manière réaliste. Ceci a motivé les studios d'effets spéciaux à adopter cette méthode de modélisation (Marvelous Designer, 2010; CreativeBloq, 2012). Une utilisation avancée requiert cependant des compétences spécifiques en conception de patrons, rendant la technique peu naturelle pour les infographistes traditionnels (CreativeBloq, 2012). De plus, un tel processus par essais et erreurs n'est pas très pratique pour reproduire avec précision le vêtement d'une référence visuelle comme une peinture, une photo ou une capture 3D.

Modélisation géométrique de vêtements

Une alternative intéressante pour concevoir des vêtements est de tirer profit des outils géométriques de modélisation 3D pour sculpter directement le vêtement sans se préoccuper de la physique sous-jacente. Deux familles d'approches sont possibles : la conception manuelle et l'acquisition géométrique automatique. D'un côté, les outils avancés d'édition de formes permettent à l'infographiste de modéliser le vêtement directement sur le personnage (Porumbescu *et al.*, 2005). Pour des vêtements relativement simples, les méthodes à base de croquis peuvent rendre le processus de conception rapide et intuitif (Igarashi et Hughes, 2003; Turquin *et al.*, 2007). Lorsqu'un vêtement est modélisé, des méthodes de transfert géométrique peuvent être utilisées pour l'adapter automatiquement à différentes morphologies, tout en préservant son style (Brouet *et al.*, 2012). D'un autre côté, grâce à l'avancée considérable de la capture en 3D ces dernières années, il est devenu possible d'acquérir avec précision la géométrie complète d'un vêtement en position statique (White *et al.*, 2007; Bradley *et al.*, 2008), permettant ainsi la création automatique de vêtements virtuels depuis une source réelle.

Les techniques géométriques sont attrayantes car elles permettent à l'utilisateur un

contrôle total sur la forme finale du vêtement. Cependant, puisque le processus de modélisation est effectué indépendamment de la physique, la forme modélisée ne peut pas être interprétée mécaniquement de manière simple. Ceci implique que le vêtement modélisé ne peut pas être animé ni modifié physiquement de manière aisée. Une approche naïve consiste à utiliser la forme comme la forme au repos du modèle physique de vêtements (qui est alors nécessairement un modèle de coque). Néanmoins, dès que la simulation démarrera, la forme s'affaissera irrémédiablement à cause de la gravité et pourra ainsi être très différente de la forme modélisée au départ, compromettant ainsi l'intérêt de la modélisation.

Modélisation inverse de vêtements

Nous pensons qu'une nouvelle manière prometteuse de modéliser des vêtements de manière réaliste et intuitive est d'investiguer une stratégie de modélisation inverse. Les problèmes inverses ont été étudiés dans beaucoup de champs scientifiques comme le génie mécanique, la météorologie, l'électromagnétisme ou encore la biologie. En génie mécanique, les problèmes inverses peuvent être classés en deux catégories (Beck et Woodbury, 1998) : (a) les problèmes de mesures inverses, où l'on cherche les propriétés du matériau et (b) les problèmes de conception inverse où c'est la configuration non déformée (la forme au repos) des objets qui est recherchée. Pour reprendre l'analogie avec le ressort dont il est question plus haut, le problème de mesure inverse (a) est celui où l'on connaît la longueur à vide ℓ_0 et où l'on cherche la masse et la raideur k, les paramètres physiques ; à l'inverse, ils sont supposés connus en conception inverse (b) et c'est le paramètre géométrique qui est l'inconnue de problème (la longueur ℓ_0).

Concernant les vêtements, les études existantes se focalisent sur le cas (a), c'est-à-dire sur l'estimation des paramètres en supposant que la forme au repos est parfaitement plate (Jojic et Huang, 1997; Bhat *et al.*, 2003; Bouman *et al.*, 2013). De telles méthodes, utiles pour identifier le type de tissu en question, sont appliquées à de simple pièces de tissu rectangulaires sous la gravité, généralement en l'absence de contact ou avec une approximation grossière (Jojic et Huang, 1997). Du fait qu'elles supposent que la forme au repos est plate, ces méthodes ne peuvent pas être utilisées pour inverser un vêtement complet (cousu) sauf si la structure exacte de l'assemblage des patrons peut être obtenue (Hasler *et al.*, 2007).

À l'inverse, dans cette thèse, notre but est de permettre à l'utilisateur d'animer n'importe quel vêtement confectionné géométriquement d'une manière pratique et consistante avec le moins d'information possible en entrée. L'utilisateur fournit seulement le maillage du vêtement qu'il peut choisir librement en fonction du type de mouvement souhaité. Comme nous ignorons délibérément la structure interne du vêtement, afin de pouvoir prendre en entrée n'importe quel maillage 3D (ne résultant pas nécessairement d'un assemblage de patrons), nous considérons que la forme au repos doit être cherchée dans l'espace 3D. Notre attention est alors portée sur les méthodes de conception inverse (b), où la forme au repos est l'inconnue.

Conception élastique inverse

Quelques approches en informatique graphique se sont récemment intéressées à l'identification de la forme au repos d'une structure élastique comme des tiges en 2D (Derouet-Jourdan *et al.*, 2010), les systèmes masse-ressort (Twigg et Kačić-Alesić, 2011) ou des éléments finis 3D (Twigg et Kačić-Alesić, 2011). Cependant, ces approches sont soit trop spécifiques, ou trop génériques pour tenir compte de la structure particulière des modèles de vêtements. De plus, aucune d'entre elles ne permet de prendre en compte le contact frottant, pourtant très présent dans les configurations que l'on observe (voir la figure $7.0.6_{94}$).

La méthode qui, dans l'esprit, est la plus proche de la nôtre est celle de la stratégie de modélisation inverse récemment introduite par Derouet-Jourdan *et al.* (2013a) dans le cas des cheveux. Dans ce travail, réalisé au sein de l'équipe Bipop de l'Inria, les auteurs montrent qu'il est possible de retrouver la forme au repos de chaque fibre telle que sous la gravité et en présence de contact frottant, la forme à l'équilibre corresponde avec celle d'un ensemble de courbes données en entrée. Un point important qui rend le problème inverse envisageable est d'utiliser un modèle (comme celui de la première partie de ce manuscrit) pour lequel les forces élastiques dépendent linéairement de la forme au repos. Le problème d'inverse statique se formule alors comme une équation linéaire en la forme au repos, soumis à des contraintes coniques dues au frottement. En utilisant des hypothèses raisonnables sur la forme au repos des cheveux pour sélectionner une solution parmi les nombreuses mathématiquement envisageables, le problème régularisé se simplifie en un programme quadratique sous contraintes coniques du second ordre (SOCCQP). Ce dernier peut alors être résolu de manière robuste et efficace avec un solveur développé, à l'origine, pour la simulation (directe) de cheveux (Daviet *et al.*, 2011).

À cause de la formulation nodale des modèles de vêtements, de fortes non-linéarités sont introduites dans les équations, contrairement à (Derouet-Jourdan *et al.*, 2013a), rendant cette méthode inutilisable dans notre cas.

Chapitre 9

Choix d'un modèle de coques élastiques

Les coques sont des solides dont deux dimensions (longueur et largeur) sont grandes devant la troisième (épaisseur). Dans le cas particulier où la position du solide au repos (soumis à aucune contrainte) est plane, on parle de plaque. Comme exemples d'objets modélisables par des plaques, on peut citer une feuille de papier — à condition de ne pas trop la déformer — ou un morceau de tissu. Pour les coques, on peut penser à un chapeau, une bouteille en plastique, une balle de tennis ou une feuille d'arbre.

La simulation de coques élastiques et plus particulièrement de plaques a très vite intéressé le domaine de l'informatique graphique, principalement pour animer des *vêtements virtuels*. En effet, les mouvements de ces surfaces étant très riches et complexes, il est important de les modéliser si l'on veut atteindre un certain réalisme. Les vêtements sont dans la plupart des cas fabriqués à partir de patrons plats cousus, leur modélisation par des plaques est donc tout à fait appropriée.

Alors, pourquoi des coques ? Nous avons choisi d'inverser des coques pour plusieurs raisons. D'une part parce que nous n'avons pas d'a priori (à part celui d'être mince) sur l'objet que nous souhaitons inverser. Celui-ci peut être un vêtement ou autre chose comme un chapeau (une coque). D'autre part, dans le cas d'un vêtement sans couture (une seule plaque), il pourra être possible de privilégier les formes au repos plates dans l'algorithme d'inversion. Enfin, soyons honnêtes, une autre raison est que ce choix permet une certaine simplification du problème. Une réelle inversion de vêtements nécessiterait de retrouver, dans la forme prise en entrée (à inverser), les patrons qui ont servi à générer celui-ci. La découpe en éléments développables (voir plus bas pour une définition dans le cas continu) a déjà occupé certains auteurs (Brouet *et al.*, 2012) mais notre problème est plus délicat. En effet, nous devrions chercher les coutures délimitant des régions planes dans l'état *non déformé* (au repos). Notons que dans certains cas, cette découpe n'est pas

unique et possède même une infinité de solutions, penser à une jupe dont le patron est un demi-anneau comme celui-ci : . Les problèmes d'inversion sont réputés difficiles, ajouter à ceci le fait que nous utilisons un modèle de contact frottant non régulier, dont nous montrons plus loin qu'il est nécessaire pour expliquer beaucoup de formes sous la gravité. Il nous a donc semblé déjà ambitieux de chercher à inverser des coques avant de considérer le cas des plaques « cousues », spécifiques aux vêtements.

Plan du chapitre Dans la section suivante, nous donnons quelques caractéristiques requises par un modèle élastique pour que son inversion soit facilitée, au moins en un certain sens. Puis, après un bref rappel sur la mécanique des coques et en passant en revue les principaux modèles de simulation utilisés en informatique graphique, nous montrons dans ce chapitre qu'un tel modèle n'a pas encore été introduit à la communauté. Pour maximiser l'impact de nos travaux dans le domaine, nous avons choisi de retenir un modèle qui semble faire référence en informatique graphique mais dont l'inversion en présence de contact frottant est un problème ouvert. Toutefois, nous donnons en fin de chapitre quelques réflexions sur la conception d'un modèle (réduit) adapté à l'inversion, travail que nous avons initié et qui est actuellement poursuivi au sein de l'équipe Bipop, par Alejandro Blumentals et Florence Bertails-Descoubes.

9.1 Difficultés de l'inversion d'un modèle élastique

L'énergie interne de la plupart des modèles élastiques discrets se met sous la forme suivante

$$E_{\text{int}}(x,\bar{x}) = \frac{1}{2}\big\langle \mathbb{K}(\bar{x})\,(\varphi(x)-\varphi(\bar{x}))\,;\varphi(x)-\varphi(\bar{x})\big\rangle \tag{9.1.1a}$$

$$= \frac{1}{2}\big\|\varphi(x)-\varphi(\bar{x})\big\|^2_{\mathbb{K}(\bar{x})}, \tag{9.1.1b}$$

où $x \in \mathbb{R}^n$ sont les coordonnées (généralisées) du modèle, $\bar{x}$ représente la forme au repos correspondante, $\mathbb{K}(\bar{x})$ est la matrice de raideur (discrète) et enfin $\varphi : \mathbb{R}^n \longrightarrow \mathbb{R}^p$ retourne les quantités intervenant dans le calcul de l'énergie. Ces quantités peuvent être par exemple une longueur ou un angle exprimés en fonction des positions. On remarque que le système étudié dans la première partie tombe dans ce formalisme (*c.f.* section $4.3.2_{60}$). Dans ce cas, x est le vecteur de courbures de la tige, φ est l'identité[1] et la matrice de raideur ne dépend pas de la forme au repos. Tout ceci en fait un modèle bien adapté à l'inversion dont les propriétés ont d'ailleurs déjà été exploitées dans le cas de Super-Hélices par Derouet-Jourdan *et al.* (2013a). *Retenons qu'un modèle élastique adapté à l'inversion est un modèle dont les forces élastiques dépendent linéairement des degrés de liberté et dont la matrice de raideur ne dépend pas de la forme au repos.* Dans le cas général, φ et $\mathbb{K}$ sont non-linéaires ce qui nous permet déjà de comprendre que le problème inverse sera difficile, même en l'absence de contact frottant.

[1] C'est ce qui permettait notamment d'impliciter les forces élastiques dans le schéma d'intégration de la dynamique.

Remarque 9.1.2. Jusqu'ici, nous avons supposé que la forme à l'équilibre était toujours connue à travers les degrés de liberté x du modèle utilisé. Cependant, il est possible que la géométrie en entrée soit dans un « format » différent de celui du modèle physique. Par exemple, dans (Derouet-Jourdan *et al.*, 2013a), les auteurs prennent en entrée des courbes discrètes (une suite de points) et les convertissent dans le « format » du modèle physique : des courbures et torsion constantes par morceaux. Cette conversion repose sur la méthode efficace mais pas directe des tangentes flottantes (Derouet-Jourdan *et al.*, 2013b). Ici, on s'intéresse aux difficultés de l'inversion du point de vue purement physique mais les problèmes de « conversions » géométriques doivent être pris en considération car un modèle facile à inverser sera probablement difficile à obtenir à partir d'un modèle nodal. Réciproquement, les modèles nodaux sont souvent difficiles à inverser mais ne nécessitent pas, dans la plupart des cas, de conversion géométrique.

Supposons que le système ne soit soumis qu'à des forces extérieures dérivant d'un potentiel que nous noterons $V(x)$. Si x est une position d'équilibre du système pour la forme au repos $\bar{x}$, on peut écrire l'équilibre des forces,

$$\nabla_x E_{\text{int}}(x, \bar{x}) + \nabla V(x) = 0. \tag{9.1.3}$$

Même si une condition de stabilité de l'équilibre pourrait être donnée, nous la laissons de côté pour l'instant. En utilisant les équations 9.1.1b$_{100}$ et 9.1.3 précédentes, il vient,

$$[D\varphi(x)]^\top \mathbb{K}(\bar{x}) \left(\varphi(x) - \varphi(\bar{x})\right) + \nabla V(x) = 0. \tag{9.1.4}$$

Souvenons-nous que l'inconnue est ici $\bar{x}$, la forme au repos, puisque nous connaissons la forme à l'équilibre x (c'est un problème inverse !). Dans le cas où φ est l'identité et où la matrice de raideur est constante, ce problème est linéaire en $\bar{x}$ et l'on comprend un des points clefs de la méthode de Derouet-Jourdan *et al.* (2013a) dans le cas des Super-Hélices. Dans le cas où φ n'est pas l'identité mais que $\mathbb{K}$ reste constante, le problème revient à « inverser » la fonction φ puisqu'il est alors linéaire en $\varphi(\bar{x})$. C'est le cas du modèle des *Discrete Elastic Rods* dont il est question dans la première partie et avec lequel nous nous sommes comparés. Dans le cas général, l'équation 9.1.4 est non-linéaire et peut être de grande taille, le problème d'inversion est donc déjà difficile même sans la présence de contact frottant. C'est pourtant celui que nous allons chercher à résoudre car il semble qu'un modèle discret de coques dont l'énergie serait linéaire en ses paramètres ne soit pas encore disponible. Cependant, nous donnons à la fin de ce chapitre des éléments de réflexion pour l'élaboration d'un tel modèle (*c.f.* section 9.5$_{108}$).

9.2 Éléments de mécanique des coques élastiques

Un modèle (continu) de coques élastiques qui permet de rendre compte des effets mécaniques dominants est celui de *Koiter* (Ciarlet, 2006, 2000). Il n'est pas le plus général car il n'est valide que sous certaines hypothèses (*c.f.* ci-dessous, notamment l'hypothèse cinématique de Kirchhoff-Love) mais il a l'avantage d'être suffisamment complet et compact : sa formulation ne fait intervenir que le déplacement de la *surface moyenne* (voir

plus loin pour une définition explicite). Sa compréhension permet de mieux appréhender la plupart des modèles que nous décrivons dans la section suivante et est à la base de nos réflexions sur l'élaboration d'un modèle réduit de coques élastiques, en fin de chapitre. C'est pourquoi nous donnons ici les détails du modèle de Koiter.

Le modèle de coques élastiques de Koiter

On considère un solide homogène — ses propriétés sont les mêmes en tout point —, isotrope — ses propriétés sont les mêmes dans toutes les directions — et élastique — le solide retrouve toujours la même forme lorsqu'il n'est plus soumis à aucune contrainte — contenu entre deux surfaces d'espacement constant $2h$ comme sur la figure 9.2.1. On appelle *surface moyenne* la surface espacée de h des surfaces extérieures ; à elle seule elle permet, dans le modèle de Koiter, de décrire toute la géométrie du solide. Le modèle de

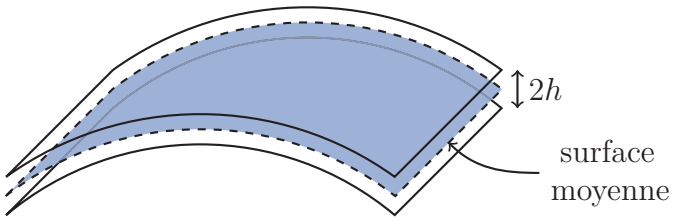


FIGURE 9.2.1 : Une coque élastique.

Koiter est bâti sur les hypothèses suivantes. Durant la déformation,

- les segments matériels orthogonaux à la surface moyenne restent droits et orthogonaux à la surface et leur longueur ne varie pas, c'est l'*hypothèse cinématique de Kirchhoff-Love*,
- les contraintes internes n'ont pas de composante normale à la surface moyenne (il n'y a pas de cisaillement interne).

Dans ces conditions, l'énergie totale de la coque peut être séparée en un terme de *membrane* et un terme de *flexion*, tous les deux satisfaisant une loi élastique non linéaire (voir l'équation $9.2.15_{105}$). Avant de pouvoir donner l'expression de cette énergie, nous introduisons les notions géométriques nécessaires à la description de la surface moyenne.

Remarque 9.2.2. Si l'on retire l'*hypothèse de Kirchhoff-Love*, en particulier, les sections ne restent pas nécessairement orthogonales à la surface moyenne au cours de la déformation — elles restent néanmoins droites —, il faut tenir compte de ceci dans le modèle. À la position de la surface moyenne, il faut ajouter une autre inconnue, le champ des *directeurs* $d : \Omega \longrightarrow \mathbb{R}^3$ qui tient compte de la déformation des sections et du changement d'épaisseur. C'est la *théorie de Cosserat avec un directeur*. On peut contraindre

l'épaisseur à ne pas varier, $d \in \mathcal{S}^2(\mathbb{R})$, c'est la *théorie spéciale de Cosserat*. Nous ne considérerons pas ces extensions dans la suite mais pour plus de détails, on pourra consulter (Antman, 2005, chapitre 17) et (Ciarlet, 1997, 2000).

Description géométrique de la surface, notions de géométrie différentielle des surfaces Dans cette section, nous nous appuyons intensivement sur (Do-Carmo, 1976) et (Ciarlet, 2006). On considère un ensemble ouvert Ω de $\mathbb{R}^2$. Notre surface est décrite par une application, $r : \Omega \ni s = (s_1, s_2) \longrightarrow r(s) \in \mathbb{R}^3$ au moins de classe $\mathscr{C}^2$ et réalisant une immersion, *i.e.*, la différentielle $Dr(s)$ est injective — $\partial_1 r(s) = \frac{\partial r}{\partial s_1}(s)$ et $\partial_2 r(s) = \frac{\partial r}{\partial s_2}(s)$ forment une famille libre — en tout point s de Ω. Cette hypothèse assure que l'on peut définir un plan tangent (et donc une normale) à la surface en tout point. Considérons $s \in \Omega$ et un déplacement infinitésimal δs tel que $s + \delta s \in \Omega$, alors

$$\|r(s+\delta s) - r(s)\|^2 = \|Dr(s)\delta s + o(\|\delta s\|)\|^2 = \left\langle \delta s\,;[Dr(s)]^\top Dr(s)\,\delta s\right\rangle + o\big(\|\delta s\|^2\big)\ ; \tag{9.2.3}$$

ceci nous montre que sur la surface, la métrique à considérer est la *première forme fondamentale*, aussi nommée le *tenseur métrique* ou encore (en élasticité) le *tenseur de déformation (à droite) de Cauchy–Green*, que nous noterons A_r, et donnée par

$$\begin{array}{rcl} A_r : \Omega & \longrightarrow & \mathcal{S}_2^{++}(\mathbb{R}) \\ s & \longmapsto & [Dr(s)]^\top Dr(s). \end{array} \tag{9.2.4}$$

Elle est toujours *symétrique définie positive* et permet de calculer des quantités métriques, comme les longueurs ou les aires, sur la surface. Par exemple, l'élément de surface infinitésimal de r en s est donné par

$$\mathrm{d}S(r,s) = \sqrt{\det(A_r(s))}\,\mathrm{d}s = \|\partial_1 r(s) \times \partial_2 r(s)\|\,\mathrm{d}s \tag{9.2.5}$$

et l'aire totale de la surface se calcule par intégration sur Ω, c'est

$$\int_\Omega \mathrm{d}S(r,s). \tag{9.2.6}$$

L'élément de longueur infinitésimal sur la surface s'écrit aussi en fonction du tenseur métrique,

$$\mathrm{d}\ell(r,s) = \sqrt{\langle \mathrm{d}s\,; A_r(s)\,\mathrm{d}s\rangle} = \|\mathrm{d}s\|_{A_r(s)}. \tag{9.2.7}$$

On dira que l'immersion r est *isométrique* lorsque $\mathrm{d}\ell(r,s) = \|\mathrm{d}s\|$, *i.e*, les chemins tracés sur Ω ont même longueur que leur image sur la surface. Il est équivalent de dire que $A_r = I$ en tout point de Ω.

Dans la suite, le tenseur métrique nous permettra de mesurer les déformations de la membrane, en cisaillement ou en élongation/compression. Afin de mesurer les déformations en flexion, il nous faut tenir compte de la variation de la normale (unitaire) à la

surface que l'on nomme *application de Gauss*[2] et que l'on définit par[3]

$$\begin{aligned} n_r : \Omega &\longrightarrow \mathcal{S}^2(\mathbb{R}) \\ s &\longmapsto \frac{\partial_1 r(s) \times \partial_2 r(s)}{\|\partial_1 r(s) \times \partial_2 r(s)\|}. \end{aligned} \tag{9.2.8}$$

Reprenons le déplacement induit par un écart δs autour de s mais regardons sa composante sur la normale maintenant,

$$\langle n_r(s)\,; r(s+\delta s) - r(s)\rangle = \big\langle n_r(s)\,; Dr(s)\delta s + \tfrac{1}{2}D^2 r(s)(\delta s, \delta s) + \mathcal{o}\big(\|\delta s\|^2\big)\big\rangle \tag{9.2.9a}$$

$$= \frac{1}{2}\big\langle n_r(s)\,; D^2 r(s)(\delta s, \delta s)\big\rangle + \mathcal{o}\big(\|\delta s\|^2\big). \tag{9.2.9b}$$

Pour mesurer les déplacements normaux à la surface, c'est donc une autre forme quadratique qu'il faut considérer, la *seconde forme fondamentale*, aussi appelée le *tenseur de courbure*, B_r, définie par

$$\begin{aligned} B_r : \Omega &\longrightarrow \mathcal{S}_2(\mathbb{R}) \\ s &\longmapsto h \mapsto \langle n_r(s)\,; D^2 r(s)(h,h)\rangle. \end{aligned} \tag{9.2.10}$$

On peut accéder à la différentielle de n_r de la manière suivante. En dérivant l'équation de l'orthogonalité de $n_r(s)$ et $Dr(s)$,

$$[Dr(s)]^\top n_r(s) = 0, \tag{9.2.11}$$

on obtient

$$B_r(s) = -[Dr(s)]^\top Dn_r(s). \tag{9.2.12}$$

Ceci signifie que la différentielle $-Dn_r(s)$, vue comme une application de l'espace tangent à r en s, $\mathrm{Im}(Dr(s))$, dans lui-même (identifiable à l'espace tangent à la sphère unité en $n_r(s)$ car tous les deux sont orthogonaux à $n_r(s)$), a pour matrice [4]

$$B_r(s)[A_r(s)]^{-1} \tag{9.2.13}$$

dans la base $Dr(s)$. Cette application permet de définir les notions classiques de courbures suivantes,

- son déterminant s'appelle la *courbure de Gauss*, notée $K_r(s)$,
- sa demi-trace définit la *courbure moyenne*, notée $H_r(s)$,
- et enfin, ses valeurs propres sont les *courbures principales* de r en s.

[2]Ce n'est pas tout à fait l'application de Gauss que nous donnons ici, l'espace de départ devrait être $r(\Omega)$ et pas simplement Ω, cette identification pose notamment des problèmes aux points doubles — où la surface passe au moins deux fois pour des paramètres différents.

[3]Cette définition est toujours possible car $\partial_1 r \times \partial_2 r$ ne s'annule pas (r est une immersion).

[4]Pour voir ceci, on peut écrire qu'un incrément δs en s induit sur l'espace tangent un incrément $\delta u = A_r(s)\delta s$; inversement, un déplacement δu sur l'espace tangent induit un déplacement $\delta s = [A_r(s)]^{-1}\,\delta u$.

Une surface est dite *développable* lorsque sa courbure de Gauss est partout nulle, *i.e.* $K_r = 0$, ceci implique que la surface est isométrique à une partie du plan. En chaque point, au moins une courbure principale est nulle, la surface ne peut donc pas être « doublement courbée » comme un paraboloïde de révolution ou une portion de sphère par exemple.

Remarque 9.2.14. Une immersion isométrique est toujours développable. Pour voir ceci, on peut dériver l'égalité $[Dr(s)]^\top Dr(s) = I$ et montrer que toutes les dérivées secondes $\partial^2_{ij} r = \frac{\partial^2 r}{\partial s_i \partial s_j}$ ont même direction que la normale n_r. En particulier, $\langle \partial^2_{11} r\,;\partial_2 r\rangle = \langle \partial^2_{12} r\,;\partial_2 r\rangle = 0$, ce qui implique après dérivation[5] que $\langle \partial^2_{11} r\,;\partial^2_{22} r\rangle = \|\partial^2_{12} r\|^2$. Or, $K_r = \det(B_r) = \langle n_r\,;\partial^2_{11} r\rangle \langle n_r\,;\partial^2_{22} r\rangle - \langle n_r\,;\partial^2_{12} r\rangle^2 = \langle \partial^2_{11} r\,;\partial^2_{22} r\rangle - \|\partial^2_{12} r\|^2 = 0$.

Énergie de Koiter Dans ce paragraphe, nous nous appuyons sur (Ciarlet, 2006) et (Ciarlet, 2000). Tous les éléments sont maintenant réunis pour que nous donnions l'expression de l'énergie interne d'une coque élastique de Koiter de surface moyenne déformée r et non déformée $\bar{r}$,

$$E^{\mathrm{K}}_{\mathrm{int}}(r,\bar{r}) = \underbrace{\frac{1}{2}\int_\Omega \frac{h}{4} \left|\!\left|\!\left| A_r(s) - A_{\bar{r}}(s) \right|\!\right|\!\right|^2_{A_{\bar{r}}(s)} \mathrm{d}S(r,s)}_{\text{énergie de membrane}} + \underbrace{\frac{1}{2}\int_\Omega \frac{h^3}{3} \left|\!\left|\!\left| B_r(s) - B_{\bar{r}}(s) \right|\!\right|\!\right|^2_{A_{\bar{r}}(s)} \mathrm{d}S(r,s)}_{\text{énergie de flexion}} \tag{9.2.15}$$

où $|\!|\!|\cdot|\!|\!|^2_{A_{\bar{r}}(s)}$ est la forme quadratique [6] associée au <u>*tenseur d'élasticité*</u> : pour $\bar{A} \in \mathcal{S}_2^{++}(\mathbb{R})$ et $M \in \mathcal{S}_2(\mathbb{R})$,

$$|\!|\!|M|\!|\!|^2_{\bar{A}} = \sum_{1\leqslant i,j,k,l\leqslant 2} M_{ij} \overbrace{\left(\frac{2E\nu}{1-\nu^2} \bar{A}_{ij}\bar{A}_{kl} + \frac{E}{1+\nu}\left(\bar{A}_{ik}\bar{A}_{jl} + \bar{A}_{il}\bar{A}_{jk}\right)\right)} M_{kl} \tag{9.2.16a}$$

$$= \frac{2E}{1+\nu}\left[\frac{\nu}{1-\nu}\Big(\mathrm{tr}\big(\bar{A}M\big)\Big)^2 + \mathrm{tr}\Big(\big(\bar{A}M\big)^2\Big)\right] \tag{9.2.16b}$$

où $E > 0$ est le *module de Young* et $\nu \in \left[-1\,;\frac{1}{2}\right]$ est le *coefficient de Poisson* du matériau considéré. Remarquons que l'énergie de Koiter mesure l'écart entre les formes fondamentales de la surface déformée r et de la surface non déformée $\bar{r}$ pour une norme qui dépend de la métrique définie sur $\bar{r}$.

Remarque 9.2.17. L'énergie de Koiter est *non-linéaire* par rapport au déplacement $r - \bar{r}$. Il existe une version *linéarisée* de l'énergie de Koiter par rapport à ce déplacement ; plus précisément, ce sont le *changement de métrique* et le *changement de courbure*, $A_r - A_{\bar{r}}$ et $B_r - B_{\bar{r}}$, qui sont linéarisés par rapport au déplacement $r - \bar{r}$ dans 9.2.15. Voir (Ciarlet, 2006, section 4.2).

[5] Nous avons besoin que r soit au moins trois fois dérivable ici.

[6] Le fait que $A_{\bar{r}}$ soit symétrique définie positive ainsi que l'hypothèse sur les coefficients de Lamé assurent que cette forme quadratique permet bien de définir une norme sur $\mathcal{S}_2(\mathbb{R})$ (*c.f.* (Ciarlet, 2006, théorème 4.4-1 p. 179)), ceci justifie la notation $|\!|\!|\cdot|\!|\!|$.

Remarque 9.2.18. Dans le cas d'une immersion *isométrique* — $A_r = I$, voir plus haut — dont la forme au repos $\bar{r}$ est plane, le terme de flexion de l'énergie de Koiter est simplement l'intégrale de la courbure moyenne au carré,

$$\frac{k}{2}\int_\Omega H_r(s)^2\,\mathrm{d}S(r,s) \quad \text{avec} \quad k = \frac{8Eh^3}{3(1-\nu^2)}. \tag{9.2.19}$$

Dans le cas d'une surface développable, cette expression est connue sous le nom d'*énergie de Willmore*.

Maintenant que les outils théoriques sont introduits, ils vont nous éclairer pour présenter les principaux modèles de coques élastiques introduits en informatique graphique.

9.3 La simulation de coques élastiques en informatique graphique

L'un des premiers travaux majeurs proposé à la communauté est probablement celui de Terzopoulos *et al.* (1987). Le gradient d'une énergie très proche[7] de celle de Koiter $9.2.15_{105}$ est approché, puis discrétisé sur une grille par différences finies. Les auteurs aboutissent ainsi à une équation différentielle ordinaire en temps qui est intégrée numériquement avec un schéma implicite-explicite (les forces élastiques ne sont pas implicitées). La double discrétisation (en espace et en temps) rend le système peu stable et les auteurs indiquent que le terme de courbure est mal conditionné. Les travaux suivants ont tenté de formuler l'énergie du système directement sur la discrétisation (le maillage) pour s'affranchir de cette double discrétisation.

Ensuite, Baraff et Witkin (1998) introduisent un modèle de plaques destiné à l'utilisation d'un schéma d'intégration en temps de type Euler implicite — le terme implicite est en fait linéarisé, ceci équivaut à ne faire que la première itération de l'algorithme de recherche de zéro de Newton–Raphson. Les énergies sont construites directement sur le modèle discret, un maillage triangulaire, auquel on associe une version non déformée plane. L'énergie de membrane est inspirée de celle de Koiter, une version discrète du tenseur métrique est calculée pour permettre la construction des termes de *cisaillement* et d'*élongation/compression*. Notons que les auteurs considèrent un terme d'élongation/compression de la forme $(\|\partial_i r\| - 1)^2$ au lieu de $\left(\|\partial_i r\|^2 - 1\right)^2$ pour que le système soit moins *raide*. Notons aussi que la comparaison avec l'énergie de Koiter montre que le terme croisé $\left(\|\partial_1 r\|^2 - 1\right)\left(\|\partial_2 r\|^2 - 1\right)$ est absent de l'énergie de membrane de Baraff et Witkin (1998). Concernant l'énergie de flexion, les auteurs utilisent une version discrétisée de l'intégrale de la courbure moyenne au carré $\int_\Omega {H_r}^2$ — *c.f.* (Cohen-Steiner et Morvan, 2003) pour une preuve de convergence —, seulement valable dans le cas d'une déformation isométrique, *c.f.* remarque 9.2.18. Cette simplification est faite dans le but

[7]Signalons que dans (Terzopoulos *et al.*, 1987), la norme utilisée pour mesurer l'écart entre les formes quadratiques n'est pas basée sur le tenseur d'élasticité ; elle n'est même pas identique pour le terme de membrane et le terme de flexion

de découpler complètement les termes de flexion et de membrane mais notons aussi que le couplage membrane/flexion — mis en évidence dans l'énergie de Koiter par l'utilisation d'une norme commune pour ces deux termes — est rompu par l'utilisation de constantes de raideur découplées.

Un modèle de coques élastiques qui est une extension de (Baraff et Witkin, 1998) est élaboré par Grinspun *et al.* (2003) (il est très proche de celui proposé la même année par Bridson *et al.* (2003)), c'est le modèle des *Discrete Shells*. Les énergies sont construites directement sur le modèle discret, un maillage triangulaire. On distingue trois termes dans l'énergie, deux termes de membrane — les quantités surmontées d'une barre sont celles de la forme au repos —,

$$E_\ell = \frac{k_\ell}{2} \sum_{\text{arêtes } e} \frac{1}{\bar{\ell}_e} \left(\ell_e - \bar{\ell}_e\right)^2 \qquad \{\ell_e \text{ est la longueur de l'arête } e\} \tag{9.3.1a}$$

$$E_a = \frac{k_a}{2} \sum_{\text{faces } f} \frac{1}{\bar{a}_f} \left(a_f - \bar{a}_f\right)^2 \qquad \{a_f \text{ est l'aire de la face } f\} \tag{9.3.1b}$$

et un terme de flexion

$$E_\theta = \frac{k_\theta}{2} \sum_{\substack{\text{arêtes} \\ \text{intérieures } e}} \frac{3\bar{\ell}_e^2}{\bar{A}_e} \left(\theta_e - \bar{\theta}_e\right)^2 \qquad \left\{\begin{matrix} A_e \text{ est la somme des aires} \\ \text{des faces adjacentes à } e \end{matrix}\right\} \tag{9.3.2}$$

où θ_e est l'angle dièdre (angle entre les normales) des triangles adjacents à l'arête e. Les nombres k_ℓ, k_a et k_θ sont des constantes de raideur dépendantes du matériau considéré. Le terme de flexion est une version discrétisée de la mesure de l'écart entre les courbures moyennes déformée et au repos. Cette expression — dans le cas d'une surface développable — est connue sous le nom d'énergie de Canham–Helfrich (Canham, 1970; Helfrich, 1973) et a été introduite empiriquement pour l'étude des *bicouches lipidiques* — les globules rouges en sont un exemple. Cette énergie est inspirée de celle de Baraff et Witkin (1998) pour une déformation isométrique (en flexion pure) de plaques — *c.f.* la remarque 9.2.18$_{106}$ — et n'est peut-être pas réaliste pour une coque extensible. Cependant, cette hypothèse est destinée à obtenir des termes de membrane et de flexion découplés pour permettre un choix des raideurs plus intuitif. Le terme de membrane est quant à lui plus difficile à justifier mathématiquement même s'il ressemble tout de même à une version discrétisée de la mesure de l'écart entre les premières formes fondamentales. Finalement, comme pour le modèle de Baraff et Witkin (1998), il n'y a pas de couplage membrane/flexion mais l'auteur précise qu'il n'est pas nécessaire de le modéliser pour de l'animation. La simplicité des termes de l'énergie en font un modèle assez facile à implémenter, pourvu que les dérivations soient faites de manière algorithmique, par calcul formel ou à la compilation en utilisant la *différentiation automatique* comme le font les auteurs (voir (Tamstorf et Grinspun, 2013) pour une discussion plus détaillée). Ceci en fait un modèle de choix pour la simulation numérique de coques élastiques. Il est assez répandu dans la communauté (Guendelman *et al.*, 2005; Wardetzky *et al.*, 2007; Umetani *et al.*, 2011).

Plus tard, les travaux se consacreront à une autre formulation du terme de flexion (Wardetzky *et al.*, 2007) ou à la limitation de l'élongation / compression (Goldenthal *et al.*, 2007; English et Bridson, 2008; Thomaszewski *et al.*, 2009).

Finalement, le modèle des *Discrete Shells* de Grinspun *et al.* (2003) apparaît comme un modèle incontournable — au moins en informatique graphique — pour la simulation numérique de coques élastiques et son inversion est un problème ouvert. C'est celui que nous choisissons d'utiliser dans la suite.

9.4 L'inversion du modèle des *Discrete Shells*

Le modèle de Grinspun *et al.* (2003) possède toutes les difficultés soulevées au début du chapitre dans la section 9.1_{100}. La fonction φ de l'équation $9.1.1\mathrm{b}_{100}$ est non linéaire — elle retourne les longueurs des arêtes, les aires des faces et les angles dièdres en fonction de la position des points du maillage — et la dépendance de la matrice de raideur $\mathbb{K}$ est elle aussi non linéaire en les positions — *c.f.* les facteurs de normalisation des énergies dans les équations $9.3.1\mathrm{a}_{107}$, $9.3.1\mathrm{b}_{107}$ et $9.3.2_{107}$.

Cependant, il pourrait être astucieux d'adapter la paramétrisation utilisée afin de rendre les dépendances linéaires. Une idée naïve consiste à utiliser les degrés de liberté ℓ_e, a_f et θ_e pour décrire la surface. L'énergie totale est alors quadratique en ces trois variables mais pas en leur équivalent au repos (ce qui nous intéresse pour l'inversion). D'autre part, il faudrait ajouter des contraintes de compatibilité (non-linéaires) car ces paramètres ne sont pas indépendants les uns des autres. En effet, on peut calculer l'aire d'un triangle en fonction de la longueur de ses côtés (c'est la formule de Héron) mais la relation est non-linéaire. Il faudrait d'autre part assurer la positivité des longueurs et des aires. Ceci semble difficile. Notons que Wang *et al.* (2012) proposent une version discrète des équations de compatibilité de Gauss et Codazzi–Mainardi (voir la section 9.5 pour une définition dans le cas continu) et donnent un algorithme pour récupérer la position des points du maillage même lorsque les quantités ne satisfont pas tout à fait les équations discrètes.

Finalement, un modèle possédant les caractéristiques décrites à la section 9.1_{100} n'a à notre connaissance pas encore été introduit à la communauté. Dans la section suivante, nous donnons un début de réflexion sur une manière possible d'élaborer un tel modèle.

9.5 Réflexions sur l'élaboration d'un modèle de coques élastiques destiné à l'inversion

L'idée d'élaborer un modèle adapté à l'inversion est fortement inspirée du succès de l'utilisation d'un modèle réduit, les Super-Hélices, pour l'inversion de coiffures dans (Derouet-Jourdan *et al.*, 2010, 2013a). L'utilisation de degrés de liberté dont l'énergie dépend quadratiquement permet de « transformer » le problème en celui du calibrage

géométrique — *geometric fitting* en anglais. En effet, l'inversion (sans contact) devient alors triviale puisqu'elle se réduit à la résolution d'un système linéaire.

Dans l'énergie de Koiter $9.2.15_{105}$, les degrés de liberté qui semblent devoir être considérés sont la première et la seconde forme fondamentale. On peut se demander si leur connaissance est suffisante pour retrouver toute la surface. La réponse est oui (au moins localement et lorsque Ω est simplement connexe), pour peu que la première forme fondamentale soit symétrique définie positive, que la seconde soit symétrique et qu'elles satisfassent des conditions de compatibilité dites de *Gauss et Codazzi–Mainardi*, *c.f.* (Do-Carmo, 1976, section 4.3, p. 234 et 235). C'est le *théorème fondamental des surfaces* aussi appelé *théorème de Bonnet* (Do-Carmo, 1976, section 4.3, p. 236). La première et la seconde forme fondamentale forment donc un bon candidat pour l'élaboration d'un modèle réduit de coques élastiques.

On sait que l'une des principales causes du mauvais conditionnement des systèmes issus de modèles discrets de plaques ou de coques élastiques est l'importante raideur de l'énergie de membrane nécessaire pour garantir une quasi-inextensibilité (Provot, 1995; Baraff et Witkin, 1998; Goldenthal *et al.*, 2007). Forts de notre expérience sur l'élaboration de modèles réduits pour les tiges, nous savons qu'il peut être intéressant de garantir cette inextensibilité directement dans la modélisation. On obtiendrait ainsi un modèle de coques élastiques en flexion pure. Pour cela, nous faisons l'hypothèse simplificatrice que la surface est une immersion isométrique[8], sa première forme fondamentale est alors partout l'identité — *c.f.* section 9.2_{101}. Ceci annule le terme de membrane de l'énergie de Koiter $9.2.15_{105}$ puisque $A_r = A_{\bar{r}} = I$. De plus, $(\partial_1 r,\, \partial_2 r,\, n_r)$ forme une base orthonormée de $\mathbb{R}^3$ — un élément de $\mathcal{SO}(3)$ — en tout point de Ω. Une rotation qui évolue continûment en fonction d'un (ici même deux) paramètre scalaire n'est pas sans nous rappeler nos travaux sur les tiges — voir le chapitre 4_{29}. Notons

$$\begin{array}{rcl} \mathcal{R} : \Omega & \longrightarrow & \mathcal{SO}(3) \\ s & \longmapsto & \mathcal{R}(s) = \big(\partial_1 r(s),\, \partial_2 r(s),\, n_r(s)\big) \end{array} \tag{9.5.1}$$

et supposons $\mathcal{R}$ de classe $\mathscr{C}^2$. On note aussi $(e_1,\, e_2,\, e_3)$ la base canonique de $\mathbb{R}^3$ de sorte que $\mathcal{R}e_1 = \partial_1 r$, $\mathcal{R}e_2 = \partial_2 r$ et $\mathcal{R}e_3 = n_r$. Enfin, le théorème $\mathrm{D}.3.1_{165}$ montre qu'il existe deux vecteurs $\mathcal{K}_1$ et $\mathcal{K}_2$, de classe $\mathscr{C}^1$, tels que[9]

$$\partial_1 \mathcal{R}(s) = \mathcal{R}(s)[\mathcal{K}_1(s)]_\times \quad \text{et} \quad \partial_2 \mathcal{R}(s) = \mathcal{R}(s)[\mathcal{K}_2(s)]_\times. \tag{9.5.2}$$

Mais ces vecteurs ne sont pas indépendants, il faut en particulier satisfaire le théorème de Schwarz pour le repère $\mathcal{R}$ et la surface r,

$$\left\{ \begin{array}{c} \partial_{12}\mathcal{R} = \partial_{21}\mathcal{R} \\ \partial_{12} r = \partial_{21} r \end{array} \right. \iff \left\{ \begin{array}{c} \partial_1 \mathcal{R}[\mathcal{K}_2]_\times + \mathcal{R}[\partial_1 \mathcal{K}_2]_\times = \partial_2 \mathcal{R}[\mathcal{K}_1]_\times + \mathcal{R}[\partial_2 \mathcal{K}_1]_\times \\ \partial_1 \mathcal{R} e_2 = \partial_2 \mathcal{R} e_1 \end{array} \right. \tag{9.5.3a}$$

$$\iff \left\{ \begin{array}{c} [\mathcal{K}_1]_\times [\mathcal{K}_2]_\times + [\partial_1 \mathcal{K}_2]_\times = [\mathcal{K}_2]_\times [\mathcal{K}_1]_\times + [\partial_2 \mathcal{K}_1]_\times \\ [\mathcal{K}_1]_\times e_2 = [\mathcal{K}_2]_\times e_1 \end{array} \right. \tag{9.5.3b}$$

[8] Cette hypothèse implique que la surface est toujours développable.

[9] Cette formulation est inspirée du modèle de Cosserat pour les tiges, voir la section $4.1.1_{30}$.

ce qui vectoriellement donne finalement

$$\langle \boldsymbol{\kappa}_1 \,;\, e_3 \rangle = \langle \boldsymbol{\kappa}_2 \,;\, e_3 \rangle = 0 \qquad \{\text{pas de rotation autour de la normale}\} \tag{9.5.4a}$$

$$\langle \boldsymbol{\kappa}_1 \,;\, e_1 \rangle = -\langle \boldsymbol{\kappa}_2 \,;\, e_2 \rangle \qquad \{\text{la surface ne subit pas de « torsion »}\} \tag{9.5.4b}$$

$$\partial_2 \boldsymbol{\kappa}_1 - \partial_1 \boldsymbol{\kappa}_2 = \boldsymbol{\kappa}_1 \times \boldsymbol{\kappa}_2. \tag{9.5.4c}$$

On peut encore raffiner un peu ces équations en posant,

$$\boldsymbol{\kappa}_1 = \begin{pmatrix} \tau \\ -\kappa_1 \\ 0 \end{pmatrix} \quad \text{et} \quad \boldsymbol{\kappa}_2 = \begin{pmatrix} \kappa_2 \\ -\tau \\ 0 \end{pmatrix}, \tag{9.5.5}$$

ce qui garantit que 9.5.4a et 9.5.4b sont satisfaites. L'équation 9.5.4c est alors équivalente à

$$\begin{cases} \kappa_1 \kappa_2 = \tau^2 & \{\text{Gauss (Do-Carmo, 1976, (5) p. 234)}\} \quad (9.5.6a) \\ \partial_2 \kappa_1 = \partial_1 \tau & \{\text{Codazzi–Mainardi (Do-Carmo, 1976, (6) p. 235)}\} \quad (9.5.6b) \\ \partial_1 \kappa_2 = \partial_2 \tau. & \{\text{Codazzi–Mainardi (Do-Carmo, 1976, (6a) p. 235)}\} \quad (9.5.6c) \end{cases}$$

Ce sont exactement les équations de compatibilité de Gauss et Codazzi–Mainardi dans le cas d'une immersion isométrique. En particulier, il n'existe pas d'autres contraintes que nous n'aurions pas encore exprimées.

Nous pouvons maintenant calculer la seconde forme fondamentale, B_r, donnée par l'expression $9.2.10_{104}$,

$$B_r = \begin{pmatrix} \langle \mathcal{R}e_3 \,;\, \partial_1 \mathcal{R}e_1 \rangle & \langle \mathcal{R}e_3 \,;\, \partial_2 \mathcal{R}e_1 \rangle \\ \langle \mathcal{R}e_3 \,;\, \partial_1 \mathcal{R}e_2 \rangle & \langle \mathcal{R}e_3 \,;\, \partial_2 \mathcal{R}e_2 \rangle \end{pmatrix} = \begin{pmatrix} -\langle \boldsymbol{\kappa}_1 \,;\, e_2 \rangle & -\langle \boldsymbol{\kappa}_2 \,;\, e_2 \rangle \\ \langle \boldsymbol{\kappa}_1 \,;\, e_1 \rangle & \langle \boldsymbol{\kappa}_2 \,;\, e_1 \rangle \end{pmatrix} \tag{9.5.7}$$

ce qui donne finalement,

$$B_r = \begin{pmatrix} \kappa_1 & \tau \\ \tau & \kappa_2 \end{pmatrix}. \tag{9.5.8}$$

Ceci nous permet de calculer les courbures, puisque $B_r[A_r]^{-1} = B_r$,

- la courbure de Gauss, $K_r = \det(B_r) = \kappa_1\kappa_2 - \tau^2 = 0$, on retrouve qu'une immersion isométrique est toujours développable ;
- la courbure moyenne, $H_r = \dfrac{\operatorname{tr}(B_r)}{2} = \dfrac{\kappa_1 + \kappa_2}{2}$;
- les courbures principales, 0 et $\rho = \kappa_1 + \kappa_2$.

Synthèse Si l'on se donne trois fonctions de Ω à valeurs réelles, $\kappa_r = (\kappa_1, \kappa_2, \tau)$, de classe $\mathscr{C}^1$ et satisfaisant le système $9.5.6_{110}$, alors il existe une unique immersion isométrique r satisfaisant $9.5.8_{110}$. De plus, on peut « reconstruire » la surface par intégration de $9.5.2_{109}$ puis $9.5.1_{109}$. D'autre part, l'énergie de Koiter, associée à une forme au repos du même type $\bar{r}$, s'écrit

$$E^{\mathrm{K}}_{\mathrm{int}}(r, \bar{r}) = \frac{1}{2}\int_\Omega \left\|\kappa_r - \kappa_{\bar{r}}\right\|^2_{\mathbb{K}} \mathrm{d}s \quad \text{avec} \quad \mathbb{K} = \frac{2Eh^3}{3(1-\nu^2)}\begin{pmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & 2(1-\nu) \end{pmatrix}. \tag{9.5.9}$$

Cette énergie satisfait les critères que nous avions relevés dans la section 9.1_{100} pour qu'un modèle élastique soit facilement inversible : la fonction φ de $9.1.1\mathrm{b}_{100}$ est l'identité et la matrice de raideur ne dépend pas de la forme au repos $\kappa_{\bar{r}}$. C'est donc un bon point de départ pour l'élaboration d'un modèle discret aisément inversible.

Remarque 9.5.10. On notera que l'expression de l'énergie interne est dans le cas présent (inextensible et sans cisaillement transverse) très proche de l'énergie élastique d'une tige de Kirchhoff (inextensible et sans cisaillement transverse) que nous avons rencontrée dans la première partie à l'équation $4.1.38_{39}$.

Difficultés relatives à l'élaboration d'un modèle discret Pour donner un modèle discret des équations ci-dessus, on pourrait par exemple choisir une famille de fonctions paramétriques — constantes par morceaux, polynomiales par morceaux, *etc* — et chercher à « projeter » sur l'espace des contraintes $9.5.6_{110}$.

On pourrait aussi exhiber une famille d'éléments de l'espace des contraintes $9.5.6_{110}$ paramétrés par des degrés de liberté bien choisis. Ceci ne semble pas évident.

Peut-être qu'une méthode à mi-chemin entre les deux précédentes — satisfaire une partie des contraintes et projeter sur les autres — serait un bon compromis. Par exemple, dans le cas où l'on contraint le vecteur κ_r à être constant, les équations $9.5.6\mathrm{a}_{110}$ et $9.5.6\mathrm{b}_{110}$ sont automatiquement satisfaites. De plus, on possède une expression explicite (de forme fermée, voir l'annexe A_{157}) pour r. Ceci fait l'objet du paragraphe suivant.

Reconstruction dans le cas de courbures constantes On suppose ici que le vecteur κ_r est constant et nous allons donner une expression explicite du repère $\mathcal{R}$ et de la surface r.

Supposons que $0 \in \Omega$ et notons $r(0) = r_0$ et $\mathcal{R}(0) = \mathcal{R}_0$. Dans ce cas, le repère s'écrit simplement en utilisant le théorème $\mathrm{D}.2.4_{164}$,

$$\mathcal{R}(s) = \mathcal{R}(s_1, 0) + \int_0^{s_2} \partial_2\mathcal{R}(s_1, v)\,\mathrm{d}v = \mathcal{R}(s_1, 0)\exp\big([s_2\mathcal{K}_2]_\times\big) \tag{9.5.11a}$$

$$= \mathcal{R}_0 \exp\big([s_1\mathcal{K}_1]_\times\big)\exp\big([s_2\mathcal{K}_2]_\times\big) \tag{9.5.11b}$$

$$= \mathcal{R}_0 \exp\big([s_1\mathcal{K}_1 + s_2\mathcal{K}_2]_\times\big) \qquad \big\{[\mathcal{K}_1]_\times \text{ et } [\mathcal{K}_2]_\times \text{ commutent.}\big\} \tag{9.5.11c}$$

L'exponentielle d'une matrice antisymétrique est donnée en annexe à l'équation D.2.5$_{164}$. On peut aussi donner l'expression de r. Pour cela, il est astucieux d'intégrer dans le système des courbures principales : d'abord sur la ligne de courbure nulle puis sur celle de courbure $\rho = \kappa_1 + \kappa_2$. Le cas de la courbure moyenne nulle — la surface est plane — est trivial et nous l'excluons ici. Ceci revient à supposer que $\rho \neq 0$ donc κ_1 et κ_2 ne peuvent pas être simultanément nuls. Quitte à permuter le rôle des indices 1 et 2, on peut supposer $\kappa_1 \neq 0$. Dans ces conditions, la famille

$$v = (v_1, v_2) = \begin{pmatrix} \kappa_1 & -\tau \\ \tau & \kappa_1 \end{pmatrix} \tag{9.5.12}$$

est une base (adaptée) de vecteurs propres de B_r. v_1 est associé à la courbure principale et v_2 à la valeur propre nulle. Le long de v_2, le repère $\mathcal{R}$ est invariant et le long de v_1, il tourne à vitesse constante $(\kappa_1+\kappa_2)$ autour de $\mathcal{R}_0 v_2$[10]. r est donc inclus dans un cylindre. Le calcul confirme ceci, grâce au changement de variable défini par

$$s = \varphi(t = (t_1, t_2)) = v\, t, \tag{9.5.13}$$

il vient

$$\mathcal{R}(\varphi(t)) = \mathcal{R}_0 \exp\big([(\boldsymbol{\kappa}_1,\, \boldsymbol{\kappa}_2)\, v\, t]_\times\big) = \mathcal{R}_0 \exp\big([-\rho t_1 v_2]_\times\big). \tag{9.5.14}$$

On peut ainsi donner l'expression de la surface par intégration du repère,

$$\begin{aligned}
r(\varphi(t)) &= r_0 + \int_0^{t_2} \mathcal{R}(\varphi(0,u)) D\varphi(0,u) e_2 \,\mathrm{d}u + \int_0^{t_1} \mathcal{R}(\varphi(u,t_2)) D\varphi(u,t_2) e_1 \,\mathrm{d}u && \text{(9.5.15a)}\\
&= r_0 + \int_0^{t_2} \mathcal{R}_0 v_2 \,\mathrm{d}u + \int_0^{t_1} \mathcal{R}(\varphi(u,t_2)) v_1 \,\mathrm{d}u && \text{(9.5.15b)}\\
&= r_0 + t_2 \mathcal{R}_0 v_2 + \int_0^{t_1} \mathcal{R}_0 \exp\big([-\rho u v_2]_\times\big) v_1 \,\mathrm{d}u && \text{(9.5.15c)}\\
&= r_0 + \mathcal{R}_0 \left(t_2 v_2 + \frac{\sin(t_1\omega)}{\omega} v_1 + \frac{1-\cos(t_1\omega)}{\rho} e_3 \right) && \text{(9.5.15d)}
\end{aligned}$$

en ayant posé $\omega = \|-\rho v_2\| = \sqrt{\rho^3 \kappa_1}$. Un exemple de tracé d'une telle surface utilisant cette dernière expression est donné figure 9.5.16$_{113}$.

Remarque 9.5.17. L'expression 9.5.15d est tout à fait similaire à celle de la courbe moyenne d'une Super-Hélice (Bertails *et al.*, 2006). Ajouté à la remarque 9.5.10$_{111}$, ceci montre le lien étroit entre ces deux modèles.

Conclusion Nous avons donné des pistes de réflexion pour l'élaboration d'un modèle réduit de coques élastiques dont l'énergie de Koiter est quadratique en fonction des paramètres (le tenseur de courbure 9.5.8$_{110}$). La tâche se révèle plus complexe que dans

[10] On utilise abusivement la même notation pour v_1 et v_2 et leur version dans $\mathbb{R}^3$ construite en ajoutant un zéro comme troisième composante.

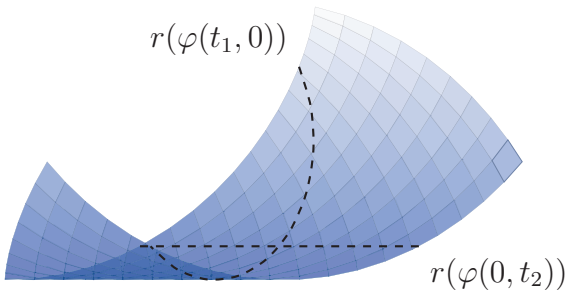


FIGURE 9.5.16 : Un exemple de surface développable à courbure moyenne constante $H_r = 1$ ($\kappa_1 = \kappa_2 = -\tau = 1$).

le cas des tiges car il faut prendre en considération des contraintes algébriques et différentielles non linéaires ($9.5.6_{110}$). Le cas des courbures constantes permet de donner une expression de forme fermée de la surface. Cependant, il semble difficile de construire une approximation d'une surface développable en recollant de tels morceaux de manière $\mathscr{C}^1$.

Ces travaux sur l'élaboration d'un modèle réduit de coques élastiques adapté à l'inversion ont été poursuivis par Alejandro Blumentals dans Blumentals *et al.* (2016).

Chapitre 10

Inversion de coques en contact frottant

Dans ce chapitre, on considère que l'on se donne (en entrée de la méthode), un maillage triangulaire conforme, discrétisation d'une surface régulière de $\mathbb{R}^3$ de sorte qu'elle puisse être simulée par le modèle des Discrete Shells que nous avons choisi au chapitre précédent. On nomme ce modèle 3D la *forme objectif*, c'est la géométrie que l'on souhaite inverser. On note aussi n_v le nombre de points que comporte le maillage et on rassemble toutes les coordonnées de ceux-ci dans le vecteur $x_\mathrm{t} \in \mathbb{R}^{3n_\mathrm{v}}$ [1].

Notre but est de trouver une forme au repos, $\bar{x}$, pour laquelle la forme objectif soit à l'équilibre sous la gravité, éventuellement en présence d'autres forces extérieures comme le contact frottant. On note E_p l'énergie potentielle du système composée de l'énergie interne donnée par les équations 9.3.1a$_{107}$ 9.3.1b$_{107}$ 9.3.2$_{107}$ auxquelles il faut éventuellement ajouter le potentiel dont dérivent les forces extérieures. Dans le cas où seules des forces dérivant d'un potentiel agissent sur le système, on cherche une forme au repos, $\bar{x}$, telle que la forme objectif soit un minimum local de l'énergie potentielle. En particulier, x_t est un point critique de cette l'énergie

$$\nabla_x E_p(x_\mathrm{t}, \bar{x}) = 0 \tag{10.0.1}$$

qui doit de plus satisfaire une condition de stabilité. Nous verrons qu'en pratique, les points critiques qui sont générés par notre algorithme sont toujours des points d'équilibre stable et nous pourrons nous contenter de l'équation 10.0.1. L'étude de ce cas fait l'objet de la section suivante. Nous montrons ensuite que la prise en compte de contraintes consistant à fixer des points du maillage peut être traitée de manière similaire au cas précédent, avec seulement quelques modifications simples. En revanche, en présence de contact frottant, le problème est plus complexe et prend une forme un peu différente de 10.0.1 comme nous le verrons dans la section 10.4$_{129}$ après une initiation au contact

[1] L'indice v signifie « vertex » (point) et l'indice t « target » (objectif).

frottant à la section 10.3_{125}. Nous montrons alors que la reformulation de la recherche des équilibres en présence de contact frottant en un problème d'optimisation permet d'utiliser, presque directement, l'algorithme mis au point dans le cas sans contact. Les résultats de la méthode sont présentés au chapitre suivant.

Ce travail a été effectué en collaboration avec Gilles Daviet, doctorant dans l'équipe Bipop et a donné lieu à la publication d'un rapport de recherche (Casati *et al.*, 2016).

10.1 Cas sans contact

Dans cette section, on examine le cas de l'inversion sans contact, aucune contrainte n'est ajoutée au système $10.0.1_{115}$. Une première approche classique mais qui échoue est détaillée, avant d'aborder la formulation robuste des moindres carrés et ses difficultés, pour aboutir à la version finale de notre algorithme d'inversion. Les notations utilisées sont celles du chapitre 9_{99}.

10.1.1 Première approche : minimisation de l'erreur d'équation d'état

Il pourrait être tentant d'attaquer le problème $10.0.1_{115}$ avec des algorithmes de recherche de zéro. Cependant, rien ne garantit qu'un tel $\bar{x}$ existe et cette formulation n'est pas très robuste à la présence de bruit dans la donnée x_{t}. On peut alors lui préférer sa version relaxée

$$\min_{\bar{x}} \frac{1}{2}\|\nabla_x E_p(x_{\mathrm{t}}, \bar{x})\|^2. \tag{10.1.1}$$

Les algorithmes de résolution envisageables ont l'avantage d'être très simples, mais malheureusement, nos tests et notre analyse montrent que ce n'est pas la bonne manière d'attaquer le problème.

En effet, si $\bar{x}^*$ est solution de 10.1.1, il n'est pas garanti que x_{t} soit un point critique de $E_p(\,\cdot\,, \bar{x}^*)$. On sait juste que la norme du gradient est minimum dans un voisinage de $\bar{x}^*$. Si on lançait la simulation en partant de x_{t} pour une forme au repos $\bar{x}^*$, l'équilibre atteint serait peut-être très éloigné de x_{t} et la solution $\bar{x}^*$ ainsi calculée n'aurait alors pas grand intérêt.

Ceci ne manque pas de se produire en pratique, en effet, 10.1.1 possède beaucoup de minima locaux et les algorithmes utilisés s'y trouvent souvent bloqués.

Enfin, même dans le cas où la solution trouvée $\bar{x}^*$ annule le gradient de l'énergie, c'est-à-dire que $\bar{x}^*$ satisfait $\nabla_x E_p(x_{\mathrm{t}}, \bar{x}^*) = 0$, rien ne garantit que x_{t} soit un minimum ni même un extremum de $E_p(\,\cdot\,, \bar{x}^*)$. De plus, il n'est pas évident de modifier la forme de l'énergie ou la formulation 10.1.1 pour guider les algorithmes vers les minima de E_p.

Le principal défaut de cette méthode est de se contraindre à chercher à annuler le gradient de l'énergie sur le sous-espace $x = x_{\mathrm{t}}$ uniquement, même si ceci implique de trouver une forme au repos pour laquelle x_{t} ne représente pas un équilibre. Pour pallier

ceci, nous présentons dans les sections suivantes un algorithme qui génère une suite de couples $(x_k, \bar{x}_k)$ qui sont toujours des équilibres et tels que la suite (x_k) se rapproche de x_t.

10.1.2 Fonction de drapé, formulation aux moindres carrés

10.1.2.1 Difficultés

Si pour chaque forme au repos $\bar{x}$ il existait une unique forme à l'équilibre x, on pourrait définir une fonction ϕ par $\phi(\bar{x}) = x$. C'est la *fonction de drapé* : elle donne la forme à l'équilibre de l'objet (la coque élastique) étant donné sa forme au repos. Pour peu que cette fonction soit bijective, la solution à notre problème s'écrirait simplement[2]

$$\bar{x}^* = \phi^{-1}(x_\mathrm{t}). \tag{10.1.2}$$

Malheureusement ces hypothèses ne sont pas satisfaites en général ; trois difficultés bien distinctes rendent le problème délicat.

Difficulté 1. Pour une forme au repos, il peut exister plusieurs équilibres : ϕ est mal définie (voir la figure $10.1.3_{118}$). En ce sens, le *problème direct est mal posé*[3]. Même en ignorant cette première difficulté, deux subsistent.

Difficulté 2. Rien ne garantit que toutes les formes à l'équilibre soient atteignables : ϕ n'est pas surjective (voir la figure $10.1.4(\mathrm{a})_{119}$).

Difficulté 3. Une même forme à l'équilibre peut être engendrée par différentes formes au repos : ϕ n'est pas injective (voir la figure $10.1.4(\mathrm{b})_{119}$).

Ce dernier point n'est pas dramatique, nous cherchons *une* forme au repos dont x_t soit un équilibre associé. Cependant, nous verrons que ceci peut devenir problématique algorithmiquement et il pourra être judicieux de « guider » les itérations pour favoriser certaines formes au repos. Favoriser les formes plates, qui sont souvent à l'origine des patrons de vêtements, s'avèrera être une stratégie payante dans certains cas.

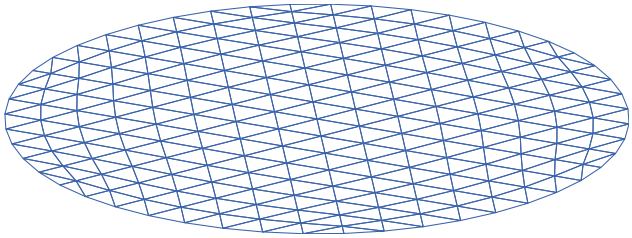
Dans les deux sections suivantes, nous montrons comment les difficultés 1 et 2 sont contournées, au moins localement.

---

[2] Le problème ainsi formulé donne son sens au terme d'*inversion*, il reprend le formalisme de la figure $7.0.4_{90}$ donné en introdction.

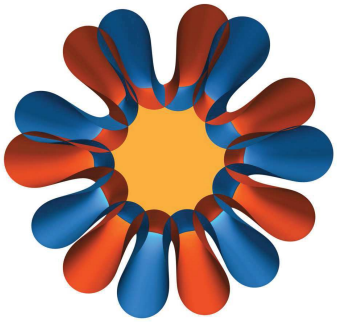
[3] Au sens de Hadamard (1902), un problème *bien posé* satisfait les trois propriétés suivantes : il existe une solution, elle est unique et dépend continuement des données. Si au moins un de ces critères n'est pas satisfait, on dit que le problème est *mal posé*.

[4] Ce maillage comporte 217 points et 384 faces, c'est la version basse résolution de celui utilisé sur les figures $10.1.3(\mathrm{b})_{118}$ et $10.1.3(\mathrm{c})_{118}$ et qui comporte 3169 points et 6144 faces.



(a) Forme au repos $\bar{x}$ (basse résolution[4])

(b) Deux équilibres, vue de côté



(c) Deux équilibres, vue de dessous

FIGURE 10.1.3 : Pour une forme au repos donnée (a), il existe plusieurs positions d'équilibre, le *problème direct* est mal posé. Elles sont superposées, l'une en bleu, l'autre en orange, sur les images (b) et (c).

10.1.2.2 Définition implicite de la fonction de drapé

Considérons un point $(x^*, \bar{x}^*)$ correspondant à un équilibre *stable* pour l'énergie potentielle E_p par rapport à la variable x et satisfaisant le critère (suffisant mais pas nécessaire, *c.f.* remarque 10.1.8$_{119}$) de stabilité du second ordre : toutes les valeurs propres de la hessienne de E_p sont strictement positives. En posant $F = \nabla_x E_p$, on peut alors écrire

$$\begin{cases} F(x^*, \bar{x}^*) = 0 & \text{(10.1.5a)} \\ F \text{ est de classe } \mathscr{C}^1 \text{ au voisinage de } (x^*, \bar{x}^*) & \text{(10.1.5b)} \\ D_x F(x^*, \bar{x}^*) \text{ est symétrique définie positive} & \text{(10.1.5c)} \end{cases}$$

ce qui permet d'appliquer le *théorème des fonctions implicites* (rappelé en annexe C$_{161}$), la condition 10.1.5c assurant l'inversibilité de la différentielle de F. On peut alors affirmer qu'il existe une unique application ϕ d'un voisinage V de $\bar{x}^*$ dans un voisinage U de x^* de classe $\mathscr{C}^1$ telle que $\forall (x, \bar{x}) \in U \times V$

$$x = \phi(\bar{x}) \Longleftrightarrow F(x, \bar{x}) = 0. \tag{10.1.6}$$

De plus, la différentielle de F par rapport à x, $D_x F$, est inversible sur $U \times V$ et celle de ϕ est donnée par

$$D\phi(\bar{x}) = -\left[D_x F(\phi(\bar{x}), \bar{x})\right]^{-1} \circ D_{\bar{x}} F(\phi(\bar{x}), \bar{x}), \quad \forall \bar{x} \in V. \tag{10.1.7}$$

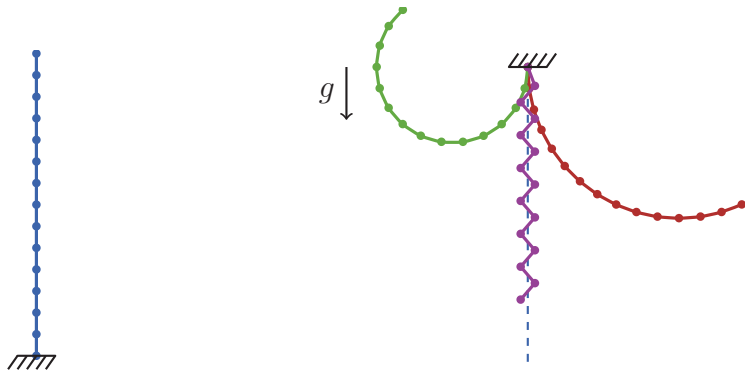


FIGURE 10.1.4 : (a) S'il n'y a pas d'énergie de flexion associée à l'encastrement (comme c'est le cas dans notre modélisation), il n'existe pas de forme au repos telle que la forme à l'équilibre ici représentée soit un équilibre stable. Ceci illustre que ϕ n'est pas surjective. (b) Sans énergie de flexion, ces trois positions au repos donnent le même équilibre (en pointillés). Ceci illustre que ϕ n'est pas injective dans le cas d'une énergie de flexion nulle. Si ce n'est pas le cas, on comprend bien que des formes à l'équilibre peuvent être proches sans que leur forme au repos le soit. Les systèmes qui sont représentés ici sont en dimension 2 mais on peut imaginer que ce sont des vues de profil d'un modèle de coques.

Une des forces du théorème des fonctions implicites est de donner une forme explicite de la différentielle de ϕ. L'équation $10.1.7_{118}$ est au cœur de la méthode de l'adjoint que nous présentons dans la section $10.1.2.3_{120}$.

Nous attirons l'attention du lecteur sur le fait que ϕ n'est pas nécessairement une bijection de V dans U, en effet, rien n'assure l'injectivité[5]. On pourrait toujours construire une surjection vers $\phi(V)$ mais rien ne dit qu'il contient x_{t}. Autrement dit, nous avons, au moins localement, contourné la difficulté 1_{117} mais les deux autres restent entières. La difficulté 2_{117} fait l'objet de la section suivante.

Remarque 10.1.8. Pour appliquer le théorème des fonctions implicites, nous avons supposé que la hessienne de E_p était symétrique définie positive en $(x^*, \bar{x}^*)$. Ceci peut sembler restrictif de prime abord car seule l'inversibilité était requise pour définir ϕ. Ce n'est pas le cas : l'inversibilité de la hessienne cumulée à la condition de stabilité de E_p assurent le caractère positif et défini de la hessienne, ces deux hypothèses sont équivalentes. Il n'en demeure pas moins que nous ne prenons pas en compte tous les équilibres : tous ceux pour lesquels la hessienne n'est pas inversible. Cependant, il semble qu'en pratique de tels points soient rares et cette hypothèse ne semble pas gêner nos algorithmes.

[5]Penser à $F(x, y) = x^2 - y$ en $(0, 0)$.

10.1.2.3 Formulation aux moindres carrés, résolution par la méthode de l'adjoint

Formulation aux moindres carrés. On relaxe la formulation 10.1.2$_{117}$ en considérant le problème suivant

$$\min_{\substack{(x,\bar{x}) \\ F(x,\bar{x})=0}} \frac{1}{2}\|x - x_{\mathrm{t}}\|^2 \tag{10.1.9}$$

Cette formulation, classique pour les problèmes inverses, a plusieurs avantages. D'une part elle est moins sensible aux erreurs dans la donnée x_{t}, d'autre part, même si le minimum trouvé ne coïncide pas avec x_{t}, c'est toujours un point d'équilibre, contrairement à la formulation 10.1.1$_{116}$, et notre algorithme garantira sa stabilité. Enfin, cette formulation permet l'utilisation des algorithmes d'optimisation qui utilisent l'information du premier ordre (gradient). En effet, elle est localement équivalente à

$$\min_{\bar{x}\in V} \underbrace{\frac{1}{2}\|\phi(\bar{x}) - x_{\mathrm{t}}\|^2}_{J(\bar{x})} \tag{10.1.10}$$

où la fonction J est $\mathscr{C}^1$ sur V et son gradient est donné par

$$\nabla J(\bar{x}) = [D\phi(\bar{x})]^\top (\phi(\bar{x}) - x_{\mathrm{t}}) . \tag{10.1.11}$$

C'est dans le calcul effectif de ce gradient qu'intervient la *méthode de l'adjoint*.

Méthode de l'adjoint. La différentielle de ϕ, donnée par 10.1.7$_{118}$, est en général pleine et son calcul nécessite une inversion de matrice, ce qui n'est pas envisageable pour les systèmes de grande taille comme ceux que nous visons. Cependant, il n'est pas nécessaire de la construire complètement pour calculer le gradient de J, seule l'évaluation de sa transposée dans la direction $\phi(\bar{x}) - x_{\mathrm{t}}$ est requise. Il suffit en effet de résoudre le système suivant, selon la *méthode de l'adjoint*,

$$\begin{cases} [D_x F(\phi(\bar{x}), \bar{x})]^\top p = x_{\mathrm{t}} - \phi(\bar{x}) & \{\textit{équation adjointe}\} \quad (10.1.12a) \\ \nabla J(\bar{x}) = [D_{\bar{x}} F(\phi(\bar{x}), \bar{x})]^\top p & (10.1.12b) \end{cases}$$

où la variable intermédiaire $p \in \mathbb{R}^{3n_{\mathrm{v}}}$ est appelé l'*état adjoint*. En se rappelant l'expression de F, on peut expliciter un peu plus ce dernier système,

$$\begin{cases} D_x^2 E_p(\phi(\bar{x}), \bar{x})p = x_{\mathrm{t}} - \phi(\bar{x}) & (10.1.13a) \\ \nabla J(\bar{x}) = D_{x,\bar{x}}^2 E_p(\phi(\bar{x}), \bar{x})p. & (10.1.13b) \end{cases}$$

Algorithme 10.1.14 – Inversion par minimisation utilisant la méthode de l'adjoint.

Données : x_{t}, $\bar{x}_0$
Résultat : x^*, $\bar{x}^*$
1 **pour** $k = 0, 1, \ldots$ **faire**
2 $\quad x_k \leftarrow \phi(\bar{x}_k)$; /* *c.f.* section 10.1.3 */
3 $\quad d_k \leftarrow \nabla J(\bar{x}_k)$; /* résoudre $10.1.13_{120}$ avec $\bar{x} = \bar{x}_k$ et $\phi(\bar{x}) = x_k$ */
4 $\quad \bar{x}_{k+1} \leftarrow$ `optim_step`$(\bar{x}_k, d_k)$;
5 **fin**

Algorithme. Il est maintenant possible d'attaquer le problème $10.1.9_{120}$ avec un algorithme de minimisation de type gradient, tous les ingrédients nécessaires étant réunis. Si on appelle `optim_step` une routine d'optimisation qui permet de passer d'un itéré au suivant grâce au gradient, on peut écrire l'algorithme 10.1.14.

Nous avons obtenu les meilleurs résultats lorsque la routine `optim_step` est celle de l'algorithme L–BFGS stabilisé avec une étape de recherche linéaire satisfaisant les conditions de Wolfe. Cet algorithme a l'avantage d'être bien adapté aux systèmes de grande taille et profite de la qualité des méthodes de type quasi-Newton.

Remarquons que la méthode de l'adjoint ne nécessite qu'*une seule* évaluation de ϕ pour le calcul de la valeur de la fonction et de son gradient. C'est la ligne 2 de l'algorithme précédent qui masque le calcul de $\phi(\bar{x}_k)$ et ses difficultés. Nous détaillons ceci dans la section suivante.

10.1.3 Procédure de drapé, appels cohérents

Nous avons vu dans la section précédente que l'algorithme auquel on aboutit impose de savoir évaluer convenablement la fonction ϕ. Celle-ci étant définie localement, parler d'une évaluation globale est abusif, c'est pourtant ce dont nous avons besoin. Dans la suite, on parlera plutôt d'une *procédure de drapé* simulant les évaluations (globales) de ϕ.

Se pose aussi le problème de la multiplicité des équilibres pour une forme au repos donnée (le problème direct est mal posé), qui rompt l'unicité de la définition de ϕ et rend difficile la cohérence des évaluations successives. Il n'est par exemple pas souhaitable que deux appels à la *procédure de drapé* pour des formes au repos proches retournent deux équilibres très éloignés. Les valeurs de la fonction J et de son gradient seraient alors incohérentes.

Dans la suite, nous construisons une *procédure de drapé* qui, à une forme au repos $\bar{x}$, associe une configuration à l'équilibre x pour laquelle les évaluations successives sont cohérentes.

10.1.3.1 Procédure de drapé

Une manière tentante de définir la *procédure de drapé* serait par la recherche d'un zéro de la fonction $F(\,\cdot\,,\bar{x})$, mais il y a plus judicieux. Il est préférable de minimiser l'énergie du système, on pourra ainsi garantir que la configuration trouvée correspond à un équilibre stable de E_p.

Pour effectuer cette minimisation, en pratique, nous utilisons l'algorithme du Newton–CG (Nocedal et Wright, 2006). C'est une méthode de Newton tronquée dans laquelle l'étape d'inversion du système est basée sur l'algorithme du gradient conjugué. La direction générée à chaque itération est garantie d'être une direction de descente, la hessienne de la fonction objectif pouvant ne pas être symétrique définie positive en des points éloignés du minimum. Cette méthode est très facile à implémenter en modifiant le gradient conjugué (que nous utilisons pour la simulation). Comme dans l'algorithme $10.1.14_{121}$, elle est couplée à une étape de recherche linéaire.

On aurait aussi pu utiliser des algorithmes plus adaptés à la recherche d'équilibres comme (Volino et Magnenat-Thalmann, 2007) mais la méthode que nous avons choisie converge très vite. En effet, elle est initialisée avec une position proche de celle de l'équilibre — on comprendra mieux pourquoi dans la section suivante — et ne nécessite en pratique qu'une dizaine d'itérations pour converger.

Malheureusement, ces précautions ne suffisent pas à garantir que ces évaluations soient cohérentes. Nous proposons une solution à ce problème dans la section suivante.

10.1.3.2 Cohérence des appels

On rappelle que l'on entend par cohérent, le fait que deux appels à la procédure de drapé sur des formes au repos proches donnent deux équilibres proches. À cause de la multiplicité des équilibres pour une forme au repos donnée, cette cohérence est difficile à satisfaire.

Il est bien évident qu'il nous faut introduire de l'information pour garantir cette cohérence et que $\bar{x}$ ne peut plus être le seul paramètre de l'algorithme. N'oublions pas qu'à terme, on souhaite obtenir une forme à l'équilibre la plus proche possible de x_t. Il est donc pertinent d'initialiser l'algorithme de minimisation avec $x_0 = x_\mathrm{t}$ mais ceci ne suffit pas.

Nous voulons en quelque sorte « éviter » les équilibres trop éloignés de x_t. Formulé comme ceci, il parait clair qu'une stratégie possible est de changer l'expression de l'énergie potentielle E_p pour la version pénalisée suivante

$$E_p{}^{\lambda}(x,\bar{x}) = E_p(x,\bar{x}) + \frac{\lambda}{2n_\mathrm{v}}\|x - x_\mathrm{t}\|^2, \quad \lambda \geqslant 0. \tag{10.1.15}$$

La difficulté revient alors à choisir convenablement le paramètre de pénalisation λ. Une valeur importante de λ tend à « convexifier » l'énergie autour de x_t, mais « tue » la dépendance en $\bar{x}$, ce qui n'est pas souhaitable. En revanche, choisir une valeur trop faible

pour le paramètre λ rend la pénalisation inutile. Il est intéressant de noter que si x_{t} est un minimum (local) de $E_p(\,\cdot\,,\bar{x})$, alors il l'est aussi pour ${E_p}^{\lambda}(\,\cdot\,,\bar{x})$. Ceci encourage à *faire décroître* la valeur du paramètre λ lorsque la confiance en $\bar{x}$ augmente.

Avant d'écrire l'algorithme complet, il nous reste à préciser avec quelle forme au repos celui-ci est initialisé. C'est l'objet de la section suivante.

10.1.4 Forme au repos initiale

Un choix pratique pour la première forme au repos de l'algorithme est $\bar{x}_0 = x_{\mathrm{t}}$ mais celle-ci a des chances d'être assez éloignée de la bonne forme au repos. Penser à l'exemple de la nappe $10.1.3_{118}$. Une autre stratégie lorsque l'on sait que la forme au repos a des chances d'être « plate » est d'utiliser une version « aplatie » de x_{t}.

Compte-tenu des outils déjà mis en place, ceci n'est pas très difficile à calculer. En effet, il suffit de minimiser l'énergie interne $E_{\mathrm{int}}(\,\cdot\,,x_{\mathrm{t}})$ en forçant les angles au repos $\bar{\theta}$ à être nuls. La minimisation va alors tendre à annuler les angles entre les faces tout en préservant (modérément) la longueur des arêtes et l'aire des faces.

10.1.5 Algorithme complet

On peut maintenant écrire l'algorithme complet de résolution du problème $10.1.9_{120}$, c'est l'algorithme 10.1.16. Noter le critère d'arrêt ligne 2. Soit la valeur de la pénalisation

Algorithme 10.1.16 – Algorithme complet d'inversion sans contact.

Données : x_{t}, λ_0, $\alpha \in \,]0\,;1[$
Résultat : $(x^*,\bar{x}^*)$ solution de $10.1.9_{120}$

1 $\bar{x}_0 \leftarrow x_{\mathrm{t}}$ ou `aplatir`(x_{t}) ; /* *c.f.* section 10.1.4 */
2 **pour** $k = 0,1,\ldots$ **jusqu'à** $\lambda_k \leqslant \varepsilon_\lambda$ ou $\|F(x_k,\bar{x}_k)\| \leqslant \varepsilon_F$ **faire**
  // recherche de $\bar{x}^{\lambda_k}$ qui minimise J^{λ_k}
3 $\bar{x}_0^{\lambda_k} \leftarrow \bar{x}_k$;
4 **pour** $i = 0,1,\ldots$ **jusqu'à** $\left\|\nabla J^{\lambda_k}\left(\bar{x}_i^{\lambda_k}\right)\right\| \leqslant \varepsilon_J$ **faire**
5 $x_i \leftarrow$ `NewtonCG`$\left({E_p}^{\lambda_k}\left(\,\cdot\,,\bar{x}_i^{\lambda_k}\right),\, x_{\mathrm{init}} = x_{\mathrm{t}}\right)$; /* *c.f.* section $10.1.3.1_{122}$ */
6 $\bar{g}_i \leftarrow$ `Adjoint`(x_i) ; /* gradient calculé par $10.1.13_{120}$ */
7 $\bar{d}_i \leftarrow$ `L-BFGS`$(\bar{g}_i)$;
8 $t_i \leftarrow$ `WolfeLS`$(t \mapsto J^{\lambda_k}(\bar{x}_i^{\lambda_k} + t\,\bar{g}_i)$; /* recherche linéaire de Wolfe */
9 $\bar{x}_{i+1}^{\lambda_k} \leftarrow \bar{x}_i^{\lambda_k} + t_i\,\bar{d}_i$;
10 **fin**
11 $\bar{x}_{k+1} \leftarrow \bar{x}^{\lambda_k}$;
12 $\lambda_{k+1} \leftarrow \alpha\lambda_k$;
13 **fin**

passe sous un certain seuil et n'est plus utile ; soit le gradient de l'énergie (non pénalisée)

est en dessous d'un certain seuil et le point x_k est stationnaire. Dans ce cas, en pratique on refait un calcul avec $\lambda = 0$ pour affiner la valeur de $\bar{x}_k$.

Nous utilisons la valeur $\alpha = \frac{1}{2}$ pour faire décroître le paramètre λ dans tous nos exemples sauf pour la nappe qui ne nécessite qu'une itération avec $\lambda_0 = 5$.

10.2 Cas des points fixés

Dans beaucoup de cas comme celui d'une nappe sur une table, d'un tissu épinglé, d'un drapeau, ou encore d'une jupe attachée à la taille, il peut être intéressant de considérer certains points du maillage comme fixes lors de l'inversion. C'est un cas particulier de *contraintes bilatérales*. La modification de l'algorithme $10.1.16_{123}$ pour prendre en compte ceci fait l'objet de cette section.

Dans le travail de Baraff et Witkin (1998), pour la simulation de vêtements, une méthode efficace et astucieuse est proposée pour contraindre les points à se déplacer librement (sans frottement) dans un espace à 2, 1 ou 0 dimensions. Une étape de filtrage, équivalente à une projection orthogonale (Ascher et Boxerman, 2003), est utilisée pour modifier l'algorithme du gradient conjugué utilisé pour la simulation. On peut montrer que celle-ci est équivalente à ajouter des contraintes bilatérales particulières au système d'équations.

Dans notre cas où l'on fixe complètement les points, la projection orthogonale utilisée, que nous notons Π, est définie comme ceci

$$[\Pi x]_i = \begin{cases} 0 & \text{si } i \text{ est fixé} \\ x_i & \text{sinon} \end{cases} \tag{10.2.1}$$

où x_i est le point d'indice i de x. L'algorithme proposé par Baraff et Witkin (1998) consiste à filtrer les multiplications par la hessienne de l'énergie avec Π dans les itérations du gradient conjugué. On montre facilement que cette astuce s'applique — en modifiant de la même manière l'algorithme du Newton–CG — aussi au calcul de l'équilibre par minimisation dans notre procédure de drapé de la section $10.1.3_{121}$. L'accès aux forces bilatérales correspondantes est alors très simple, ce sont les composantes non nulles du gradient de l'énergie (non filtré) évalué au point d'équilibre. Dans le chapitre 11_{141} exposant nos résultats, on présente ces forces sur la figure $11.1.1(\mathrm{a})_{144}$.

Ce qui est plus surprenant encore c'est que pour tenir compte de ces contraintes dans la méthode de l'adjoint, il suffit d'utiliser (encore une fois) l'algorithme de Baraff et Witkin (1998) lors de l'inversion de l'équation adjointe de $10.1.13_{120}$. Ceci est expliqué en fin de chapitre dans le complément $10.\mathrm{A}.1_{135}$.

Le contact frottant est-il vraiment nécessaire ? Puisque les points en contact sont connus au début de l'inversion et qu'ils seront de toute façon fixés par le frottement statique, il est légitime de se demander s'il ne serait pas possible d'utiliser la méthode des points fixés (détaillée plus haut) pour gérer le cas du contact frottant. Malheureusement,

nous montrons avec les résultats du chapitre 11_{141} que les deux approches ne sont pas équivalentes. En effet, une force associée à un contact bilatéral peut être orientée de manière arbitraire, à l'extérieur du cône de Coulomb (*c.f.* section 10.3.1) par exemple ou même faire un angle obtus avec la normale au contact, auquel cas il ne s'agit même plus d'une force de contact unilatéral mais d'une force d'adhésion. Voir en particulier la figure $11.1.1_{144}$.

10.3 Intermède : initiation au contact frottant

Le frottement mécanique entre les objets physiques est à l'origine de beaucoup de phénomènes visuellement riches et que l'informatique graphique cherche à modéliser. On peut penser aux effets de groupement dans les cheveux (Daviet *et al.*, 2011) ou aux plis d'un vêtement, sur une manche retroussée par exemple, comme sur la figure $7.0.6_{94}$. Il est donc important de tenir compte du contact frottant dans la modélisation afin de reproduire ces phénomènes. Ceci devient crucial pour l'inversion car la forme d'un objet à l'équilibre est souvent en bonne partie expliquée grâce au contact et au frottement (statique), comme le montre la figure $7.0.6_{94}$.

La modélisation du contact frottant est un problème complexe car elle cherche à rendre compte d'effets non-réguliers comme les phénomènes à seuil[6]. Il existe dans la littérature plusieurs modèles de contact frottant mais celui qui est largement adopté est le modèle de *frottement de Coulomb* que nous présentons dans la section suivante. Ce modèle fait référence car il offre un bon compromis entre simplicité (pour que les calculs numériques soient envisageables) et réalisme physique (pour rendre compte des effets dominants).

Les outils d'analyse convexe ont prouvé leur efficacité dans le traitement des problèmes de contact frottant (Acary et Brogliato, 2008; Daviet *et al.*, 2011; Derouet-Jourdan *et al.*, 2013a) et nous les utilisons dans nos travaux d'inversion en présence de contact frottant. Nous consacrerons la fin de cette section à l'introduction de ces outils.

10.3.1 Le modèle de frottement de Coulomb

Nous introduisons ici succinctement la loi de Coulomb. Pour plus de détails, on pourra consulter (Acary et Brogliato, 2008). Considérons deux objets A et B en contact en un point I à un instant donné. On suppose que la surface des objets est suffisamment régulière pour pouvoir définir une normale au point de contact, elle est notée e et dirigée de B vers A. La figure $10.3.1_{126}$ résume la situation. On note r la force exercée en I par le corps B sur le corps A et u la vitesse relative de A par rapport à B (en I). On

[6]Un phénomène a seuil issu du contact frottant simple à visualiser est celui d'une boite sur un plan dont la pente augmente. Pour les pentes faible, le frottement retient la boite, mais pour une valeur critique de l'angle de la pente, la boite va se mettre à glisser.

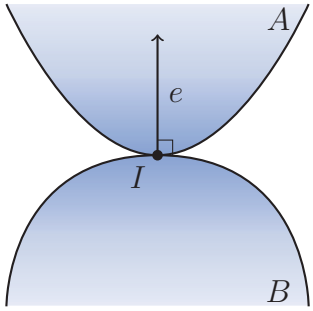


FIGURE 10.3.1 : Deux objets en contact ponctuel et la normale au contact.

se donne aussi un *coefficient de frottement*, $\mu \geqslant 0$. Il nous reste à introduire le *cône de frottement* pour pouvoir énoncer la loi de Coulomb (voir la figure 10.3.4).

Définition 10.3.2. Pour $e \in \mathbb{R}^n$ non nul et $\mu \geqslant 0$, on définit le *cône de frottement* (ou *cône du second ordre*) par,

$$K(e,\mu) = \{x \in \mathbb{R}^n \,/\, \|x_{\mathrm{T}}\| \leqslant \mu x_{\mathrm{N}}\}, \tag{10.3.3}$$

où $x_{\mathrm{N}} = \left\langle x\,;\frac{e}{\|e\|}\right\rangle$ et $x_{\mathrm{T}} = x - x_{\mathrm{N}}\frac{e}{\|e\|}$ désignent respectivement les composantes normale et tangentielle de x par rapport à e.

Le modèle de Coulomb distingue trois configurations (mutuellement exclusives) pour les vecteurs u et r :

- décollage : $r = 0$ et $u_{\mathrm{N}} > 0$,
- adhérence : $r \in K(e,\mu)$ et $u = 0$,
- glissement : $\|r_{\mathrm{T}}\| = \mu r_{\mathrm{N}}$, $r \neq 0$, $u_{\mathrm{N}} = 0$ et $u_{\mathrm{T}} = -\lambda r_{\mathrm{T}}$ avec $\lambda > 0$.

Ceci est schématisé sur la figure 10.3.4.

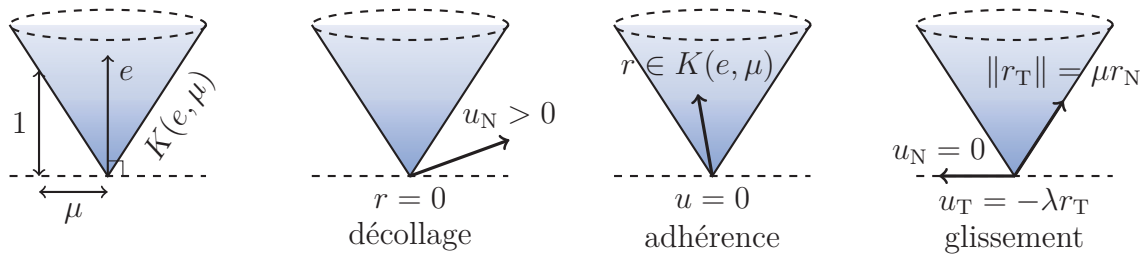


FIGURE 10.3.4 : Le cône de frottement $K(e,\mu)$ et les trois configurations de la loi de Coulomb.

Remarque 10.3.5. Dans le cas où les objets sont à l'équilibre, toutes les vitesses sont nulles (on parle d'état *statique*) et le système est forcément en adhérence. Ceci impose que la force de contact r appartienne au cône de frottement. C'est ce cas qui nous intéressera tout particulièrement dans la suite car pour l'inversion, nous cherchons des configurations à l'équilibre.

10.3.2 Outils d'analyse convexe

Les définitions et résultats que nous introduisons ici sont détaillés dans (Hiriart-Urruty et Lemaréchal, 1993). On rappelle qu'une partie convexe C est un ensemble qui contient un segment dès lors qu'il contient ses extrémités.

Définition 10.3.6. On introduit le *cône normal* à une partie fermée convexe C de $\mathbb{R}^n$ en $x \in C$,

$$\mathcal{N}_C(x) = \{d \in \mathbb{R}^n \,/\, \langle d\,;c-x\rangle \leqslant 0,\ \forall c \in C\}. \tag{10.3.7}$$

C'est l'ensemble des directions qui forment un angle obtus avec tous les vecteurs joignant x et un point de C. Un exemple est donné sur la figure 10.3.8. Lorsque x est

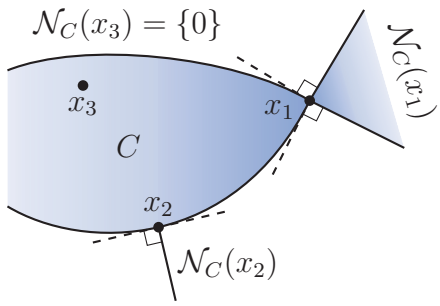


FIGURE 10.3.8 : Un exemple d'une partie convexe et fermée C et quelques-uns de ses cônes normaux.

à l'intérieur de C (dans $\mathring{C}$), $\mathcal{N}_C(x)$ est réduit au singleton $\{0\}$. Si x est un point du contour de C, deux cas se produisent. Si ce point de contour est régulier, alors le cône normal contient (le demi espace vectoriel engendré par) la normale (sortante) à C en ce point. Sinon, c'est tout un cône de directions qui sont dites *normales* à C en x.

Cette notion est assez sympathique avec le produit cartésien de convexes fermés.

Propriété 10.3.9. *On se donne $p \geqslant 1$ convexes fermés $C_i \subset \mathbb{R}^n$ et on considère le convexe (lui aussi fermé) $C = \prod_{i=1}^p C_i$. Pour $x = (x_1, \ldots, x_p) \in C$, on a*

$$\mathcal{N}_C(x) = \prod_{i=1}^{p} \mathcal{N}_{C_i}(x_i). \tag{10.3.10}$$

Dit de manière mnémotechnique : « le cône normal d'un produit est le produit des cônes normaux ».

dem. Soit $d \in \prod_{i=1}^{p} \mathcal{N}_{C_i}(x_i)$, et $c \in C$, $\langle d\,;c-x\rangle = \sum_{i=1}^{p} \langle d_i\,;c_i - x_i\rangle \leqslant 0$, on en conclut que $\prod_{i=1}^{p} \mathcal{N}_{C_i}(x_i) \subset \mathcal{N}_C(x)$. Prenons maintenant $d \in \mathcal{N}_C(x)$ et $1 \leqslant i \leqslant p$ et montrons que $d_i \in \mathcal{N}_{C_i}(x_i)$. Soit $c_i \in C_i$, le vecteur $c = (x_1, \ldots, x_{i-1}, c_i, x_{i+1}, \ldots, x_p)$ étant dans C, $\langle d\,;c-x\rangle = \langle d_i\,;c_i - x_i\rangle \leqslant 0$, ce qui montre que $d_i \in \mathcal{N}_{C_i}(x_i)$ et finalement $\mathcal{N}_C(x) \subset \prod_{i=1}^{p} \mathcal{N}_{C_i}(x_i)$. L'égalité est finalement démontrée. □

Le cône normal va surtout nous servir à donner des *conditions d'optimalité* comme le montre le résultat d'analyse convexe suivant.

Théorème 10.3.11. *Si f est une application différentiable d'un ouvert U de $\mathbb{R}^n$ dans $\mathbb{R}$, et que $x^* \in U \cap C$ est un minimum local de f sur l'ensemble convexe C, alors l'opposé du gradient de f en x^* est* normal *à C en x^** i.e.*,*

$$-\nabla f(x^*) \in \mathcal{N}_C(x^*). \tag{10.3.12}$$

Cette condition est suffisante lorsque f est convexe.

dem. Raisonnons par l'absurde. Si $-\nabla f(x^*) \notin \mathcal{N}_C(x^*)$, c'est qu'il existe $c \in C$ tel que $\langle \nabla f(x^*)\,;c-x^*\rangle < 0$. Cela entraîne que $\lim_{t\to 0^+} \frac{1}{t}\left(f\left(x^* + t(c-x^*)\right) - f(x^*)\right) < 0$. C'est donc qu'il existe $t^* \in \,]0\,;1[$ tel que $\forall t \in \,]0\,;t^*]$, $f\left(x^* + t(c-x^*)\right) < f(x^*)$. Il y a donc dans C et aussi près que l'on veut de x^*, des points dont l'image par f est strictement inférieure à $f(x^*)$. Ceci contredit le fait que x^* soit un minimum local de f sur C.

Montrons la réciproque dans le cas où f est convexe. Pour $x \in U$, $f(x) \geqslant f(x^*) + \langle \nabla f(x^*)\,;x-x^*\rangle$ (f convexe). Dans le cas où $x \in U \cap C$, $-\nabla f(x^*) \in \mathcal{N}_C(x^*)$ implique $\langle \nabla f(x^*)\,;x-x^*\rangle \geqslant 0$ et donc $f(x) \geqslant f(x^*)$. Cette dernière inégalité étant vraie dans un voisinage de x^* relatif à C, x^* est bien un minimum local de f sur C. □

Dans la suite, nous aurons besoin des deux définitions suivantes dont une schématisation est donnée figure 10.3.15.

Définition 10.3.13. Étant donné un cône $K(e,\mu)$ (*c.f.* 10.3.2$_{126}$) avec $\mu > 0$, on définit son *cône dual*, $K^\star(e,\mu)$, et son *cône polaire*, $K^\circ(e,\mu)$, par

$$K^\star(e,\mu) = K\left(e, \frac{1}{\mu}\right) \quad \text{et} \quad K^\circ(e,\mu) = -K^\star(e,\mu) = K^\star(-e,\mu) = K\left(-e, \frac{1}{\mu}\right). \tag{10.3.14}$$

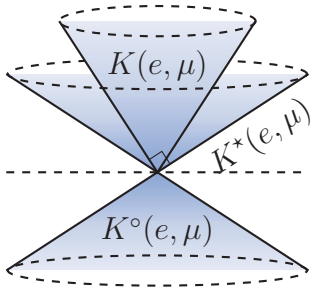


FIGURE 10.3.15 : Les cônes dual et polaire du cône $K(e,\mu)$, dans $\mathbb{R}^3$.

Dans les applications qui nous intéresseront, l'ensemble C du théorème précédent sera toujours un cône (ou un produit de cônes) $K(e,\mu)$. Le théorème suivant donne la forme du cône normal dans le cas où $C = K(e,\mu)$.

Théorème 10.3.16. *Pour $e \in \mathbb{R}^n$ non nul, $\mu > 0$ et $x \in K(e,\mu)$, le cône normal au cône $K(e,\mu)$ en x est donné par*

$$\mathcal{N}_{K(e,\mu)}(x) = K^\circ(e,\mu) \cap \{x\}^\perp . \tag{10.3.17}$$

---

dem. Sans perdre de généralité, on supposera dans la suite de cette démonstration que e est normé. Soit $d \in \mathcal{N}_{K(e,\mu)}(x)$. 0 étant dans $K(e,\mu)$, $\langle d\,;-x\rangle \leqslant 0$. De la même manière, $2x$ étant dans $K(e,\mu)$ (c'est un cône), $\langle d\,;x\rangle \leqslant 0$. Finalement, $\langle d\,;x\rangle = 0$ et d est bien un élément de $\{x\}^\perp$. Posons $c = x + d_{\mathrm{T}} + \frac{\|d_{\mathrm{T}}\|}{\mu} e$ (on envoie d sur le cône le long de e). $c \in K(e,\mu)$ car $c_{\mathrm{N}} = x_{\mathrm{N}} + \frac{\|d_{\mathrm{T}}\|}{\mu}$, $c_{\mathrm{T}} = x_{\mathrm{T}} + d_{\mathrm{T}}$ et $\|c_{\mathrm{T}}\|^2 - \mu^2 {c_{\mathrm{N}}}^2 \leqslant 2(\langle x_{\mathrm{T}}\,;d_{\mathrm{T}}\rangle - \mu x_{\mathrm{N}}\|d_{\mathrm{T}}\|) \leqslant 0$ et c_{N} est positif. On peut alors écrire $\langle d\,;c-x\rangle \leqslant 0$ qui entraîne que $\|d_{\mathrm{T}}\|^2 + d_{\mathrm{N}}\frac{\|d_{\mathrm{T}}\|}{\mu} \leqslant 0$ et d est bien dans $K^\circ(e,\mu)$. Ceci étant vrai pour tout d dans $\mathcal{N}_{K(e,\mu)}(x)$, on a montré la première inclusion $\mathcal{N}_{K(e,\mu)}(x) \subset K^\circ(e,\mu) \cap \{x\}^\perp$. Prenons maintenant $d \in K^\circ(e,\mu) \cap \{x\}^\perp$ et $c \in K(e,\mu)$. $\langle d\,;c-x\rangle = \langle d\,;c\rangle = \langle d_{\mathrm{T}}\,;c_{\mathrm{T}}\rangle + d_{\mathrm{N}}c_{\mathrm{N}} \leqslant \|d_{\mathrm{T}}\|\|c_{\mathrm{T}}\| + d_{\mathrm{N}}c_{\mathrm{N}} \leqslant c_{\mathrm{N}}(\mu\|d_{\mathrm{T}}\| + d_{\mathrm{N}}) \leqslant 0$. La seconde inclusion est vérifiée, ceci achève la démonstration. □

Finalement, on peut donner le principal théorème que nous utiliserons dans la suite qui donne les conditions d'optimalité dans le cas $C = K(e,\mu)$.

Théorème 10.3.18. *Pour $e \in \mathbb{R}^n$ non nul, $\mu > 0$ et f une application différentiable d'un ouvert U de $\mathbb{R}^n$ dans $\mathbb{R}$, si $x^* \in U \cap K(e,\mu)$ est un minimum local de f sur $K(e,\mu)$ alors la* condition de complémentarité conique du second ordre *suivante est satisfaite*

$$K^\star(e,\mu) \ni \nabla f(x^*) \perp x^* \in K(e,\mu). \tag{10.3.19}$$

Cette condition devient suffisante quand f est convexe.

---

dem. x^* étant dans $K(e,\mu)$ par hypothèse, d'après le théorème 10.3.11$_{128}$, il suffit de montrer que $-\nabla f(x^*) \in \mathcal{N}_{K(e,\mu)}(x^*) \Longleftrightarrow x^* \perp \nabla f(x^*) \in K^\star(e,\mu)$ ce qui est évident d'après le théorème 10.3.16. □

10.4 Prise en compte du contact frottant

Nous avons vu dans la section précédente qu'une des difficultés de la prise en compte du contact tient au fait que l'équilibre n'est plus expliqué seulement par la forme au repos $\bar{x}$ mais il faut aussi tenir compte de nouvelles inconnues, les forces de contact.

Dans le cas des points fixés, nous avons finalement réussi à nous affranchir du calcul de ces forces en formulant le problème uniquement sur les positions. C'est aussi ce que nous allons faire ici même si le problème est beaucoup plus délicat, les forces de contact devant satisfaire une loi non régulière.

L'équation d'équilibre s'écrit maintenant

$$\nabla_x E_p(x, \bar{x}) = f_{\mathrm{c}} \tag{10.4.1}$$

où le membre de droite contient les forces de contact frottant. Dans la suite, nous faisons l'hypothèse (simplificatrice) que notre maillage est suffisamment fin pour que seuls des contacts aux nœuds soient considérés. De plus, les seuls points en contact que nous autorisons pour les différentes formes x_k générées pendant l'algorithme sont ceux de la forme à inverser x_{t}. Ceci implique que les points en contact de x_{t} sont les mêmes que ceux de la forme à l'équilibre x^* trouvée à la fin de l'algorithme. Nous discutons ces hypothèses dans la section 10.4.3$_{133}$.

On note I_{c} l'ensemble des indices des points de x_{t} qui sont en contact. Lorsque $i \in I_{\mathrm{c}}$, la composante i du vecteur f_{c}, $f_{\mathrm{c}i}$, représentant la force de contact appliquée au point i, doit être dans le cas d'adhérence de la loi de Coulomb (*c.f.* remarque 10.3.5$_{127}$), c'est-à-dire, $f_{\mathrm{c}i} \in K(e_i, \mu_i)$ où e_i et μ_i sont respectivement la normale et le coefficient de frottement associés au contact i. Si i n'est pas dans I_{c}, le point est à l'équilibre, sans contact, et $f_{\mathrm{c}i}$ doit être nulle.

Finalement, si on note

$$\mathcal{K} = \prod_{i=1}^{n_{\mathrm{v}}} \mathcal{K}_i \quad \text{avec} \quad \mathcal{K}_i = \begin{cases} K(e_i, \mu_i) & \text{si } i \in I_{\mathrm{c}} \\ \{0\} & \text{sinon,} \end{cases} \tag{10.4.2}$$

le problème inverse se formule ainsi,

$$\text{trouver } \bar{x}^* \text{ tel que } \nabla_x E_p(x_{\mathrm{t}}, \bar{x}^*) \in \mathcal{K}. \tag{10.4.3}$$

C'est un problème non-linéaire sous contraintes coniques qui n'a en général pas qu'une seule solution. Comme dans la section 10.1.2.3$_{120}$, nous préférons relaxer ce problème par rapport à la variable x, pour une version aux moindres carrés, plus robuste. Mais ceci pose deux questions.

Comment obtenir une procédure de drapé ? On pourrait modifier la formulation 10.4.3 dans le but de définir une procédure de drapé, en s'inspirant de la section 10.1.3$_{121}$, comme ceci

$$\text{étant donné } \bar{x}, \text{ trouver } x \text{ tel que } \nabla_x E_p(x, \bar{x}) \in \mathcal{K}. \tag{10.4.4}$$

Sous cette forme, la formulation n'est pas très adaptée à un calcul numérique. En effet, on pourrait chercher à résoudre un problème du type

$$\text{étant donné } \bar{x}, \text{ trouver } x \text{ tel que } G(\nabla_x E_p(x, \bar{x})) = 0 \tag{10.4.5}$$

où G est telle que $G(y) = 0 \Longleftrightarrow y \in \mathcal{K}$. Nous montrons (dans un autre contexte) dans le complément 10.A.2$_{136}$ que ceci implique que la différentielle de G soit nulle sur $\mathcal{K}$; plus précisément, c'est la différentielle de G_i qui est nécessairement nulle sur $\mathcal{K}_i$ pour $i \in I_{\mathrm{c}}$. Le conditionnement de la différentielle de G va donc très probablement se dégrader

à l'approche de $\mathcal{K}$ et la méthode sera globalement peu stable. Dans la section 10.4.1, nous montrons comment on peut identifier $10.4.4_{130}$ à un problème de minimisation sous contraintes coniques dont la résolution par l'algorithme $10.4.13_{132}$ est mieux adaptée à un traitement numérique.

Comment généraliser la méthode de l'adjoint ? En notant $F = \nabla_x E_p$, comme dans la section $10.1.2.2_{118}$, on peut relaxer le problème $10.4.3_{130}$ en s'inspirant de la formulation aux moindres carrés $10.1.9_{120}$ comme ceci

$$\min_{\substack{(x,\bar{x}) \\ F(x,\bar{x}) \in \mathcal{K}}} \frac{1}{2}\|x - x_{\mathrm{t}}\|^2. \tag{10.4.6}$$

Cette formulation n'est pas directement compatible avec la méthode de l'adjoint que nous utilisions jusqu'ici. Cependant, dans la section $10.4.2_{133}$, nous transformons la condition d'appartenance au cône $\mathcal{K}$ en une égalité pour laquelle l'algorithme de l'adjoint peut être envisagé.

10.4.1 Le drapé comme un problème d'optimisation

Afin de transformer la formulation $10.4.4_{130}$ en un problème d'optimisation sous contraintes, mieux adapté au traitement numérique, on cherche une fonction convexe f et un ensemble, lui aussi convexe, C tels que le problème $\min_C f$ ait pour conditions d'optimalité $\nabla_x E_p(x, \bar{x}) \in \mathcal{K}$. On montre que l'hypothèse f convexe ne peut pas être satisfaite dans le cas général. Un choix pertinent est de prendre $f(h) = E_p(x_{\mathrm{t}} + h, \bar{x})$ (non convexe) et $C = \mathcal{K}^\star$ (convexe) avec

$$\mathcal{K}^\star = \prod_{i=1}^{n_{\mathrm{v}}} \mathcal{K}_i^\star \quad \text{pour} \quad \mathcal{K}_i^\star = \begin{cases} K^\star(e_i, \mu_i) & \text{si } i \in I_{\mathrm{c}} \\ \mathbb{R}^3 & \text{sinon} \end{cases} \tag{10.4.7}$$

où $K^\star(e_i, \mu_i) = K\left(e_i, \frac{1}{\mu_i}\right)$ est le cône dual du cône $K(e_i, \mu_i)$ (*c.f.* définition $10.3.13_{128}$). Dans la section $10.4.3_{133}$, nous discutons les avantages et inconvénients de cette formulation. Le problème d'optimisation considéré est

$$\min_{h \in \mathcal{K}^\star} E_p(x_{\mathrm{t}} + h, \bar{x}). \tag{10.4.8}$$

Le cône normal à $\mathcal{K}^\star$ en $h \in \mathcal{K}^\star$ est donné par la propriété $10.3.9_{127}$ et le théorème $10.3.16_{129}$,

$$\mathcal{N}_{\mathcal{K}^\star}(h) = \prod_{i=1}^{n_{\mathrm{v}}} \mathcal{N}_{\mathcal{K}_i^\star}(h_i) = \prod_{i=1}^{n_{\mathrm{v}}} (-\mathcal{K}_i) \cap \mathcal{H}_i(h_i) = -(\mathcal{K} \cap \mathcal{H}(h)) \tag{10.4.9}$$

où l'on a posé

$$\mathcal{H}(h) = \prod_{i=1}^{n_{\mathrm{v}}} \mathcal{H}_i(h_i) \quad \text{avec} \quad \mathcal{H}_i(h_i) = \begin{cases} \{h_i\}^\perp & \text{si } i \in I_{\mathrm{c}} \\ \mathbb{R}^3 & \text{sinon.} \end{cases} \tag{10.4.10}$$

Finalement, on peut écrire les conditions nécessaires (et suffisantes dans le cas convexe) d'optimalité de $10.4.8_{131}$ pour un point x de $x_\mathrm{t} + \mathcal{K}^\star$,

$$\nabla_x E_p(x, \bar{x}) \in \mathcal{K} \cap \mathcal{H}(x - x_\mathrm{t}). \tag{10.4.11}$$

Cette condition n'est pas tout à fait la même que $10.4.4_{130}$ mais regardons de plus près. Puisque $\mathcal{K} \cap \mathcal{H}(h) \subset \mathcal{K}$, si h est solution de $10.4.8_{131}$, alors $x = x_\mathrm{t} + h$ est solution de $10.4.4_{130}$. Résoudre $10.4.8_{131}$ est donc a priori restrictif dans le sens où certains points peuvent satisfaire le problème $10.4.4_{130}$ sans correspondre à un minimum de $10.4.8_{131}$. Cependant, cette approche présente plusieurs avantages : elle fournit un algorithme de drapé $10.4.8_{131}$ et elle est motivée par le cas convexe. En effet, dans ce cas, $\bar{x}^*$ solution de $10.4.3_{130}$ est équivalent à 0 solution de $10.4.8_{131}$ pour $\bar{x} = \bar{x}^*$. Mais l'intérêt le plus important de cette méthode est que lorsque les composantes en contact de x sont les mêmes que celles de $\bar{x}$ — autrement dit, $(I - \Pi)(x - x_\mathrm{t}) = 0$ —, résoudre $10.4.4_{130}$ est exactement équivalent à résoudre 10.4.11.

Pour résoudre $10.4.8_{131}$ numériquement, et ainsi obtenir notre *procédure de drapé*, un algorithme de type gradient (conjugué) projeté peut être envisagé. En effet, il est relativement simple de projeter sur les contraintes de $10.4.8_{131}$: la projection orthogonale d'une direction h sur $\mathcal{K}^\star$ est donnée par

$$[\Pi_{\mathcal{K}^\star}(h)]_i = \Pi_{\mathcal{K}_i^\star}(h_i) = \begin{cases} \Pi_{K^\star(e_i,\mu_i)}(h_i) & \text{si } i \in I_\mathrm{c} \\ h_i & \text{sinon.} \end{cases} \tag{10.4.12}$$

où $\Pi_{K^\star(e_i,\mu_i)}$ est la projection orthogonale sur le cône $K^\star(e_i, \mu_i)$. Elle est explicitement donnée, ainsi que sa différentielle, dans (Cadoux, 2009, annexe E). Nous pouvons finalement écrire la procédure de drapé, c'est l'algorithme 10.4.13. Remarquer que les

Algorithme 10.4.13 – Algorithme du gradient projeté pour l'évaluation du drapé en présence de contact frottant.

Données : $\bar{x}$, $\alpha \in]0\,;1[$
Résultat : x solution de $10.4.8_{131}$

```
x_0 ← x_t;
pour k = 0, 1, ... faire
    d_k ← ∇_x E_p(x_k, x̄);
    ρ ← 1;
    répéter
        x_{k+1} ← x_t + Π_{K*}(x_k − x_t − ρ d_k);
        ρ ← αρ;
    jusqu'à E_p(x_{k+1}, x̄) < E_p(x_k, x̄);
fin
```

stratégies adoptées dans le cas sans contact sont tout à fait transposables ici : on peut initialiser la minimisation avec $x_0 = x_\mathrm{t}$ — c'est d'ailleurs le cas dans l'algorithme 10.4.13

— et il est facile de « guider » les itérations en pénalisant l'énergie potentielle comme dans la section $10.1.3_{121}$ en modifiant l'énergie potentielle comme dans $10.1.15_{122}$.

Cette procédure de drapé est essentielle dans la méthode de l'adjoint mais nous avons vu que la formulation $10.4.6_{131}$ n'était pas directement envisageable pour cette méthode. Dans la section suivante, nous modifions cette formulation pour permettre le calcul du gradient et rendre ainsi possible la résolution du problème $10.4.6_{131}$ avec un algorithme d'optimisation.

10.4.2 Il faut sauver l'adjoint

En notant $\Pi_{\mathcal{K}}$ la projection orthogonale sur $\mathcal{K}$, on remarque que l'appartenance à $\mathcal{K}$ peut être exprimée par une égalité à zéro,

$$x \in \mathcal{K} \Longleftrightarrow x - \Pi_{\mathcal{K}}(x) = 0. \tag{10.4.14}$$

En effet, ceci est équivalent à $x_i = 0$ pour $i \notin I_{\mathrm{c}}$ et $\Pi_{K(e_i,\mu_i)}(x_i) = x_i \Longleftrightarrow x_i \in K(e_i, \mu_i)$ lorsque $i \in I_{\mathrm{c}}$. En posant

$$G = F - \Pi_{\mathcal{K}} \circ F, \tag{10.4.15}$$

le problème $10.4.6_{131}$ est équivalent à la formulation suivante,

$$\min_{\substack{(x,\bar{x}) \\ G(x,\bar{x})=0}} \frac{1}{2}\|x - x_{\mathrm{t}}\|^2. \tag{10.4.16}$$

Cependant, la méthode de l'adjoint telle que nous l'avons utilisée plus haut n'est pas envisageable ici. En effet, G n'est même pas différentiable puisque $\Pi_{\mathcal{K}}$ ne l'est pas sur le bord de $\mathcal{K}$; néanmoins, toutes les dérivées directionnelles existent, voir (Zarantonello, 1971, p. 300) ou (Hiriart-Urruty et Lemaréchal, 1993, proposition 5.3.5 p. 141). Dans l'annexe $10.\mathrm{A}.2_{136}$, nous montrons qu'un autre choix pour G, différentiable mais non inversible (c'est de toute façon impossible, *c.f.* $10.\mathrm{A}.2_{136}$), permet de donner le gradient du problème 10.4.16 et ainsi envisager sa résolution.

Nous avons détaillé toutes les étapes nécessaires à la prise en compte du contact frottant. L'algorithme est finalement proche de celui énoncé sans contact, $10.1.16_{123}$. Seul les étapes des lignes 5 et 6 sont effectuées différemment. Le résultat de la ligne 5, correspondant à la procédure de drapé, est maintenant calculé avec l'algorithme $10.4.13_{132}$. Celui de la ligne 6, qui correspond au calcul du gradient par la méthode de l'adjoint est donné dans l'annexe $10.\mathrm{A}.2_{136}$.

10.4.3 Discussion

Nous discutons ici quelques inconvénients de notre approche.

Nous avons supposé que les points en contact de l'équilibre trouvé sont ceux de la forme objectif ce qui n'est peut-être pas le cas. En effet, il est possible que l'équilibre trouvé soit en contact avec les objets de la scène car leur géométrie n'est jamais prise en

compte dans la méthode (à part pour le calcul des points de contact de la forme objectif). Cependant, si la méthode converge (équilibre et forme objectif sont confondus), les points en contact seront bien ceux de la forme objectif.

Revenons quelques instants sur le choix de la procédure de drapé qui a été effectué dans la section $10.4.1_{131}$. La formulation sur laquelle elle repose apporte l'avantage important d'être un problème d'optimisation dont la résolution peut être envisagée par des méthodes classiques. Cependant, elle repose sur l'hypothèse (importante) que l'équilibre x^* correspondant à la forme au repos trouvée à la fin de l'algorithme vérifie

$$(I - \Pi)(x^* - x_{\mathrm{t}}) = 0. \tag{10.4.17}$$

C'est-à-dire que les positions des points en contact sont les mêmes pour l'équilibre et la forme objectif. Ceci est motivé par le fait que, si l'inversion a réussi, équilibre et forme objectif sont confondus. Si l'équation 10.4.17 n'est pas vérifiée, deux cas se produisent pour les points (en contact) de la position d'équilibre qui sont distincts de leur homologue de la forme objectif. D'après la condition d'optimalité $10.4.11_{132}$, soit la composante du gradient est nulle et la position du point en question n'est en fait pas expliquée par le contact mais par l'équilibre élastique uniquement, ce qui n'est pas très gênant. Soit elle est sur le bord du cône de Coulomb et orthogonal à $x - x_{\mathrm{t}}$, lui-même sur le cône polaire en ce point. Ceci a pour effet de « bloquer » artificiellement le point en question et celui-ci risque de se déplacer lors de la simulation : il n'était pas dans une position d'équilibre. En pratique, ceci ne se produit que rarement et l'équilibre finalement trouvé par le système lors de la simulation reste très proche de la forme objectif.

Nous donnons maintenant les résultats obtenus par la méthode dont il a été question dans ce chapitre.

Compléments

10.A Modification de l'adjoint pour l'ajout de contraintes

Dans cette annexe nous détaillons la modification à apporter au système de la méthode de l'adjoint 10.1.13$_{120}$ d'abord pour la prise en compte des points fixés (section 10.2$_{124}$) puis dans le cas contact frottant (section 10.4$_{129}$).

10.A.1 Cas des points fixés

On note n_{c} le nombre de points fixés et on reprend la définition de l'opérateur Π du système 10.2.1$_{124}$: il envoie les composantes fixées sur 0 et laisse les autres inchangées. Notons que cet opérateur est auto-adjoint. L'idée de la formalisation que nous exposons ici est de « supprimer » du système les degrés de libertés inutiles, on construit alors un système plus petit. Pour cela, on considère l'opérateur $\hat{\Pi}$ de $\mathbb{R}^{3n_{\mathrm{v}}}$ dans $\mathbb{R}^{3(n_{\mathrm{v}}-n_{\mathrm{c}})}$ qui supprime les composantes fixées. On remarque que l'opérateur « inverse », qui à un vecteur de $\mathbb{R}^{3(n_{\mathrm{v}}-n_{\mathrm{c}})}$ associe un élément de $\mathbb{R}^{3n_{\mathrm{v}}}$ en ajoutant des 0 aux composantes fixées, est $\hat{\Pi}^{\top}$. Enfin, l'égalité $\hat{\Pi}^{\top} \circ \hat{\Pi} = \Pi$ nous sera utile.

Le système peut être paramétré par les seuls « vrais » degrés de liberté $\hat{x} \in \mathbb{R}^{3(n_{\mathrm{v}}-n_{\mathrm{c}})}$ qui permettent de « reconstruire » la variable

$$x = \hat{\Pi}^{\top}\hat{x} + (I - \Pi)\, x_{\mathrm{t}} \in \mathbb{R}^{3n_{\mathrm{v}}}. \tag{10.A.1}$$

La nouvelle énergie que nous considérons est donc

$$\hat{E}_p : (\hat{x}, \bar{x}) \longmapsto E_p\Big(\hat{\Pi}^{\top}\hat{x} + (I - \Pi)\, x_{\mathrm{t}}, \bar{x}\Big) \tag{10.A.2}$$

dont le gradient par rapport à $\hat{x}$, $\hat{F}$, est donné par

$$\hat{F} : (\hat{x}, \bar{x}) \longmapsto \hat{\Pi} F\Big(\hat{\Pi}^{\top}\hat{x} + (I - \Pi)\, x_{\mathrm{t}}, \bar{x}\Big). \tag{10.A.3}$$

Enfin, puisque $\left\|\hat{\Pi}^{\top}\hat{x} + (I - \Pi)\, x_{\mathrm{t}} - x_{\mathrm{t}}\right\| = \left\|\hat{x} - \hat{\Pi} x_{\mathrm{t}}\right\|$, le problème 10.1.9$_{120}$ dans le cas des points fixés s'écrit,

$$\min_{\substack{(\hat{x},\bar{x}) \\ \hat{F}(\hat{x},\bar{x})=0}} \frac{1}{2}\left\|\hat{x} - \hat{\Pi} x_{\mathrm{t}}\right\|^2. \tag{10.A.4}$$

On constate aisément que $\hat{F}$ est différentiable et l'on peut alors appliquer la méthode de l'adjoint telle qu'énoncée dans le cas non contraint au nouveau système réduit. Calculons les différentielles, $D_{\hat{x}}\hat{F}$ et $D_{\bar{x}}\hat{F}$, intervenant dans le système $10.1.12_{120}$,

$$D_{\hat{x}}\hat{F} = \hat{\Pi} \circ D_x F \circ \hat{\Pi}^\top \tag{10.A.5a}$$

$$D_{\bar{x}}\hat{F} = \hat{\Pi} \circ D_{\bar{x}} F. \tag{10.A.5b}$$

L'analogue du système $10.1.12_{120}$ permettant le calcul du gradient par la méthode de l'adjoint en un point (non projeté) $(x, \bar{x})$ s'écrit (on omet les dépendances en $(x, \bar{x})$),

$$\left\{ \begin{array}{rcl} \left[D_{\hat{x}}\hat{F}\right]^\top \hat{p} & = & \hat{\Pi}\,(x_\mathrm{t} - x) \\ \nabla J(\bar{x}) & = & \left[D_{\bar{x}}\hat{F}\right]^\top \hat{p} \end{array} \right. \iff \left\{ \begin{array}{rcl} \hat{\Pi}\,[D_x F]^\top \Pi p & = & \hat{\Pi}\,(x_\mathrm{t} - x) \\ \nabla J(\bar{x}) & = & [D_{\bar{x}} F]^\top \Pi p \end{array} \right. \tag{10.A.6a}$$

$$\iff \left\{ \begin{array}{rcl} \Pi\,[D_x F]^\top \Pi p & = & \Pi\,(x_\mathrm{t} - x) \\ \nabla J(\bar{x}) & = & [D_{\bar{x}} F]^\top \Pi p \end{array} \right. \tag{10.A.6b}$$

où p est tel que $\hat{\Pi} p = \hat{p}$. Pour résoudre ce dernier système, il suffit de résoudre l'équation adjointe de $10.1.13_{120}$ avec l'algorithme de projection de Baraff et Witkin (1998) — dont il est question dans la section 10.2_{124} — puis d'effectuer la deuxième partie du calcul avec l'état adjoint (Πp) ainsi trouvé. Notons que cet état adjoint a toujours des composantes nulles aux points fixés, il est dans l'image de Π.

Remarque 10.A.7. La seconde équation de 10.A.6b n'est pas projetée sur l'image de Π ce qui montre que le gradient de J peut avoir des composantes non nulles pour des indices correspondants à des points fixés. Au cours de l'algorithme d'inversion, les coordonnées de $\bar{x}$ correspondant à des indices fixés peuvent tout à fait changer (c'est le cas dans l'exemple de la nappe du chapitre 11_{141} des résultats, même si ce n'est pas visible).

10.A.2 Cas du contact frottant

En présence de contact frottant, on cherche à résoudre le problème $10.4.6_{131}$,

$$\min_{\substack{(x,\bar{x}) \\ G(x,\bar{x})=0}} J(x) = \frac{1}{2}\|x - x_\mathrm{t}\|^2 \tag{10.A.8}$$

pour une fonction G de la forme $H_\mathcal{K} \circ F$ et $H_\mathcal{K} : \mathbb{R}^{3n_\mathrm{v}} \longrightarrow \mathbb{R}^{3n_\mathrm{v}}$ satisfaisant

$$x \in \mathcal{K} \iff H_\mathcal{K}(x) = 0. \tag{10.A.9}$$

Le but de cette section est de donner une méthode de calcul du gradient de J par rapport à la variable $\bar{x}$, nécessaire à la résolution de $10.4.6_{131}$, que nous noterons de manière (très) abusive $\nabla_{\bar{x}} J$, afin de calquer les notations de la section $10.1.2.3_{120}$.

Choix de la fonction $H_{\mathcal{K}}$ Discutons maintenant le choix de $H_{\mathcal{K}}$. Afin que G soit différentiable aux points générés par l'algorithme de drapé — ces points $(x, \bar{x})$ satisfont $F(x, \bar{x}) \in \mathcal{K}$, voir la section 10.4.1$_{131}$ —, il est nécessaire que $H_{\mathcal{K}}$ soit différentiable sur $\mathcal{K}$. Pour simplifier, supposons que $H_{\mathcal{K}}$ agit composante par composante, plus précisément,

$$[H_{\mathcal{K}}(x)]_i = H_{\mathcal{K}_i}(x_i), \quad x \in \mathbb{R}^{3n_{\mathrm{v}}},\ i \in [\![1\,;n_{\mathrm{v}}]\!]. \tag{10.A.10}$$

Un choix canonique pour les indices qui ne sont pas en contact ($\mathcal{K}_i = \{0\}$) est

$$H_{\mathcal{K}_i} = I \quad \text{pour} \quad i \notin I_{\mathrm{c}}, \tag{10.A.11}$$

leur différentielle sera alors l'identité. Il est intéressant de constater que le choix des applications $H_{\mathcal{K}_i}$ pour $i \in I_{\mathrm{c}}$ n'a guère d'influence sur la différentielle de G. En effet, comme le montre le théorème suivant, leur différentielle en $x \in \mathcal{K}_i = K(e_i, \mu_i)$ sera toujours nulle.

Théorème 10.A.12. *Pour C un convexe de $\mathbb{R}^n$ d'intérieur non vide et un entier $p \geqslant 1$, s'il existe une fonction $f : \mathbb{R}^n \longrightarrow \mathbb{R}^p$ différentiable sur C telle que*

$$x \in C \Longleftrightarrow f(x) = 0, \tag{10.A.13}$$

alors la différentielle de f en un point de C est l'application nulle.

---

dem. Fixons $x \in C$. Il existe une base $(h_1, \ldots, h_n)$ de $\mathbb{R}^n$ telle que $x + h_i \in C$, $\forall i \in [\![1\,;n]\!]$. En effet, si ce n'était pas le cas, $C - x$ (*resp.* C) serait inclus dans un espace vectoriel (*resp.* affine) de dimension strictement inférieure à n et donc serait d'intérieur vide, ce qui contredit les hypothèses faites sur C. Pour $i \in [\![1\,;n]\!]$ et $t \in]0\,;1]$, $x + th_i$ appartient à C (C est convexe) et $f(x + th_i) = f(x) = 0$. Donc $\lim_{t \to 0^+} \frac{1}{t}(f(x + th_i) - f(x)) = 0 = Df(x)h_i$. Finalement, $Df(x)$ étant nulle sur une base de $\mathbb{R}^n$, c'est bien l'application nulle. □

Remarquons que ce théorème ne s'applique avec $C = K(e, \mu)$ que pour $\mu > 0$, sinon l'intérieur de C est vide.

Il reste encore à s'assurer qu'une telle application f existe bien, c'est l'objet du théorème qui suit.

Théorème 10.A.14. *Pour $e \in \mathbb{R}^n$ non nul, $\mu > 0$ et un entier $p \geqslant 1$, il existe une application $f : \mathbb{R}^n \longrightarrow \mathbb{R}^p$ différentiable sur $K(e, \mu)$ (et même $\mathscr{C}^\infty$ sur $\mathbb{R}^n$) telle que*

$$x \in K(e, \mu) \Longleftrightarrow f(x) = 0. \tag{10.A.15}$$

---

dem. Nous allons démontrer ce résultat dans le cas $p = 1$, le cas général s'en déduisant simplement en ajoutant des composantes nulles à f. Dans la suite, on suppose $\|e\| = 1$. Considérons l'application $g : \mathbb{R} \longrightarrow \mathbb{R}_+$ telle que $g(t) = \exp(-1/t^2)$ pour $t < 0$ et $g(t) = 0$ sinon, de sorte que $g(t) = 0 \Longleftrightarrow t \geqslant 0$. On vérifie aisément que g est de classe $\mathscr{C}^\infty$ sur $\mathbb{R}$. On pose $f : \mathbb{R}^n \longrightarrow \mathbb{R}$, $x \longmapsto g(x_{\mathrm{N}}) + g\big(\mu^2 {x_{\mathrm{N}}}^2 - \|x_{\mathrm{T}}\|^2\big)$. On vérifie que f est bien candidate : $f(x) = 0 \Longleftrightarrow \big(x_{\mathrm{N}} \geqslant 0 \text{ et } \mu^2 {x_{\mathrm{N}}}^2 \geqslant \|x_{\mathrm{T}}\|^2\big) \Longleftrightarrow x \in K(e, \mu)$. Enfin, f est $\mathscr{C}^\infty$ sur $\mathbb{R}^n$ en tant que composée de fonctions $\mathscr{C}^\infty$. □

Finalement, en choisissant les composantes $H_{\mathcal{K}_i}$ pour $i \in I_\mathrm{c}$ comme au théorème précédent et comme indiqué en 10.A.11$_{137}$ pour $i \notin I_\mathrm{c}$, la différentielle de $H_\mathcal{K}$ s'écrit

$$DH_\mathcal{K}(x) = \Pi, \quad \forall x \in \mathcal{K} \tag{10.A.16}$$

où Π est définie en 10.2.1$_{124}$: elle envoie les composantes en contact sur 0 et laisse les autres inchangées.

Calcul des différentielles de G On peut maintenant calculer les différentielles de G, D_xG et $D_{\bar{x}}G$, en un point $(x, \bar{x})$ satisfaisant $F(x, \bar{x}) \in \mathcal{K}$,

$$D_xG(x, \bar{x}) = DH_\mathcal{K}(F(x, \bar{x})) \circ D_xF(x, \bar{x}) = \Pi \circ D_xF(x, \bar{x}). \tag{10.A.17}$$

De la même manière,

$$D_{\bar{x}}G(x, \bar{x}) = \Pi \circ D_{\bar{x}}F(x, \bar{x}). \tag{10.A.18}$$

À ce stade, on comprend qu'il n'est pas possible d'utiliser directement le théorème des fonctions implicites comme précédemment puisqu'il est clair que la composition à droite par Π dans 10.A.17 rend la différentielle de G par rapport à x non inversible.

Calcul du gradient de J On se donne un point $(x, \bar{x})$ qui annule G et un déplacement, amené à tendre vers 0, $(\delta x, \delta\bar{x})$ qui nous laisse sur la contrainte $G = 0$. Le fait de rester sur la contrainte impose

$$D_xG(x, \bar{x})\delta x + D_{\bar{x}}G(x, \bar{x})\delta\bar{x} = \mathcal{o}(\delta x, \delta\bar{x}). \tag{10.A.19}$$

À ce stade, nous faisons l'hypothèse importante suivante, que nous discutons plus loin,

$$x - x_\mathrm{t} \in \mathrm{Im}\Big([D_xG(x, \bar{x})]^\top\Big), \tag{H*}$$

ceci revient à supposer l'existence d'un vecteur p, tel que

$$[D_xF(x, \bar{x})]^\top \Pi p = x - x_\mathrm{t}. \tag{10.A.20}$$

Nous pouvons maintenant exprimer l'incrément sur J en fonction de $\delta\bar{x}$ uniquement.

$$\begin{aligned}
J(x + \delta x) - J(x) &= \langle x - x_\mathrm{t}\,;\delta x\rangle + \mathcal{o}(\delta x, \delta\bar{x}) && \text{(10.A.21a)}\\
&= \langle p\,; D_xG(x, \bar{x})\delta x\rangle + \mathcal{o}(\delta x, \delta\bar{x}) && \text{(10.A.21b)}\\
&= \langle p\,; -D_{\bar{x}}G(x, \bar{x})\delta\bar{x}\rangle + \mathcal{o}(\delta x, \delta\bar{x}) && \text{(10.A.21c)}\\
&= \big\langle \underbrace{-[D_{\bar{x}}G(x, \bar{x})]^\top p}_{\nabla_{\bar{x}}J(x,\bar{x})}\,;\delta\bar{x}\big\rangle + \mathcal{o}(\delta x, \delta\bar{x}), && \text{(10.A.21d)}
\end{aligned}$$

et le calcul de $\nabla_{\bar{x}}J(x, \bar{x})$ est donné par la résolution du système suivant,

$$\begin{cases}
[D_xF(x, \bar{x})]^\top \Pi p = x - x_\mathrm{t} & \text{(10.A.22a)}\\
\nabla_{\bar{x}}J(x, \bar{x}) = -[D_{\bar{x}}F(x, \bar{x})]^\top \Pi p. & \text{(10.A.22b)}
\end{cases}$$

Malheureusement, l'hypothèse $\mathrm{H}^*{}_{138}$ n'est pas toujours satisfaite. Remarquons cependant que notre méthode de drapé suggère de conserver les composantes en contact c'est-à-dire $x_i = x_{\mathrm{t}i}$ pour $i \in I_{\mathrm{c}}$ ou encore $x - x_{\mathrm{t}} \in \mathrm{Im}(\Pi)$ (voir la section $10.4.1_{131}$). Si nous cherchons à satisfaire ceci dans l'inversion, on est amené à contraindre δx à être dans $\mathrm{Im}(\Pi)$. Ceci mène exactement au système $10.\mathrm{A}.6\mathrm{b}_{136}$ auquel nous aboutissons lorsque nous fixons les points et qui peut être résolu avec la méthode de Baraff et Witkin (1998). Une autre approche consiste à résoudre aux moindres carrés, on aboutit au système suivant,

$$\begin{cases} \Pi D_x F(x,\bar{x}) \left[D_x F(x,\bar{x})\right]^\top \Pi p = \Pi D_x F(x,\bar{x})(x - x_{\mathrm{t}}) & \text{(10.A.23a)} \\ \nabla_{\bar{x}} J(x,\bar{x}) = -\left[D_{\bar{x}} F(x,\bar{x})\right]^\top \Pi p, & \text{(10.A.23b)} \end{cases}$$

qui est lui aussi du type de $10.\mathrm{A}.6\mathrm{b}_{136}$ que nous savons résoudre.

Chapitre 11

Résultats

11.1 Contact bilatéral *versus* contact unilatéral avec frottement

Dans la section 10.2_{124}, nous avons présenté une méthode pour tenir compte du fait que certains points du maillage sont fixés. Il était apparu tentant de traiter le cas du contact frottant de cette manière. Cependant, comme le montre la figure $11.1.1_{144}$ dans le cas 2D, ceci échoue. En effet, les forces de contact unilatéral (les force nécessaires à fixer les points) peuvent être dirigées vers l'obstacle ou être hors du cône (fictif) de Coulomb. En revanche, notre méthode, lorsqu'elle utilise le frottement de Coulomb, parvient à modifier la forme aplatie (initialisant l'algorithme) en la resserrant et ainsi ramener les forces de contact dans le cône de Coulomb.

11.2 Validation

Pour tester notre méthode, nous avons considéré trois modèles.

(i) Une nappe que nous avions déjà présentée figure $10.1.3_{118}$. Ici, on connaît exactement la forme au repos, elle est plate et représentée figure $10.1.3(\mathrm{a})_{118}$. La forme objectif est donnée par l'équilibre d'une simulation où les points touchant la table sont fixés comme expliqué section 10.2_{124}. Les paramètres physiques utilisés pour l'inversion sont ceux de la simulation : $k_\ell = 5 \cdot 10^{-1}$, $k_a = 5 \cdot 10^{-2}$, $k_\theta = 2{,}5 \cdot 10^{-5}$ (*c.f.* section 9.3_{106}) et une masse surfacique $\rho = 100\,\mathrm{g/m^2}$. Le maillage comporte 3169 points dont 817 sont fixés (à la table). Cet exemple semble « facile » puisqu'on est sûr qu'une solution existe. Cependant, la difficulté réside ici dans la multiplicité des équilibres (le problème direct est mal posé) dont deux sont exposés figure $10.1.3_{118}$. Ajoutons que la forme initiale utilisée est une version « aplatie » de la forme objectif avec l'algorithme donné dans la section $10.1.4_{123}$.

(ii) Le second modèle est une manche ballon réalisée par un infographiste[1] qui a utilisé plusieurs méthodes : un peu de simulation physique (pour un modèle et des paramètres inconnus), et des outils de modélisation géométrique comme la FFD[2]. Les paramètres physiques utilisés pour l'inversion sont ceux d'un vêtement ni vraiment rigide ni vraiment souple : $k_\ell = 3 \cdot 10^{-2}$, $k_a = 3 \cdot 10^{-3}$, $k_\theta = 5 \cdot 10^{-6}$. Le maillage comporte 2045 points dont 102 sont en contact avec le bras et fixés par la méthode décrite dans la section 10.2_{124}. L'inversion est initialisée avec la forme objectif. La difficulté de cet exemple réside dans le fait que l'effet « bouffant » est difficile à obtenir physiquement. En effet, si l'on simule ce modèle par la méthode naïve, celui-ci s'affaisse et se stabilise dans une position qui n'est pas très stable, et qui peut même devenir complètement aplatie si l'on perturbe le système (*c.f.* figures $11.2.2_{146}$ et $11.2.3_{146}$). Ici, un modèle de coques élastiques est clairement nécessaire.

(iii) Le dernier modèle est une jupe, elle aussi réalisée par un infographiste[3] à base de patrons cousus et simulés (avec un modèle et des paramètres inconnus) sous la gravité sur un personnage que nous utilisons ensuite. Pour ce modèle, nous utilisons le contact frottant. Le maillage comporte 5208 points dont 755 en contact. Les paramètres du modèle du simulateur de vêtements utilisés pour l'inversion sont les mêmes que ceux de la manche ballon et le coefficient de frottement est $\mu = 0{,}3$. L'inversion est initialisée avec la forme objectif. La difficulté de cet exemple réside d'une part dans le traitement du contact frottant et d'autre part dans le fait qu'aucun point n'est fixé. La seule manière de faire tenir la jupe sur le personnage, c'est grâce au frottement. Une simulation par « inversion naïve » (forme au repos = forme objectif) de ce modèle est donnée figure $11.2.1_{145}$.

Les résultats sont donnés figure $11.2.2_{146}$.

On peut voir que le cas de la nappe ne pose aucun problème à notre algorithme. La forme au repos utilisée pour générer la forme à l'équilibre est même retrouvée avec une erreur très faible. Ici, ce qui aide l'algorithme à ne pas tomber dans le piège des équilibres multiples, et ainsi contourner le caractère mal posé du problème direct, vient très certainement de deux choses. Il y a en effet deux composantes dans notre algorithme qui permettent de privilégier l'équilibre le plus proche de la forme objectif : l'initialisation de la procédure de drapé par la forme objectif (*c.f.* section $10.1.3.1_{122}$) et la pénalisation des formes trop éloignées des la forme objectif (*c.f.* section $10.1.3.2_{122}$).

L'exemple de la manche ballon semble être plutôt bien traité par notre approche. S'il est vrai que la forme à l'équilibre correspondante à la forme au repos retrouvée est proche de la forme objectif, elle n'est pas très stable. En effet, l'ajout d'un champ de force (pour simuler du vent par exemple) fait se « dégonfler » le maillage. Ceci est visible figure $11.2.3_{146}$. La stabilité de l'équilibre est pourtant garantie par notre approche mais

[1] Il s'agit de Laurence Boissieux, ingénieur de recherche à l'Inria Grenoble–Rhône-Alpes, merci à elle.

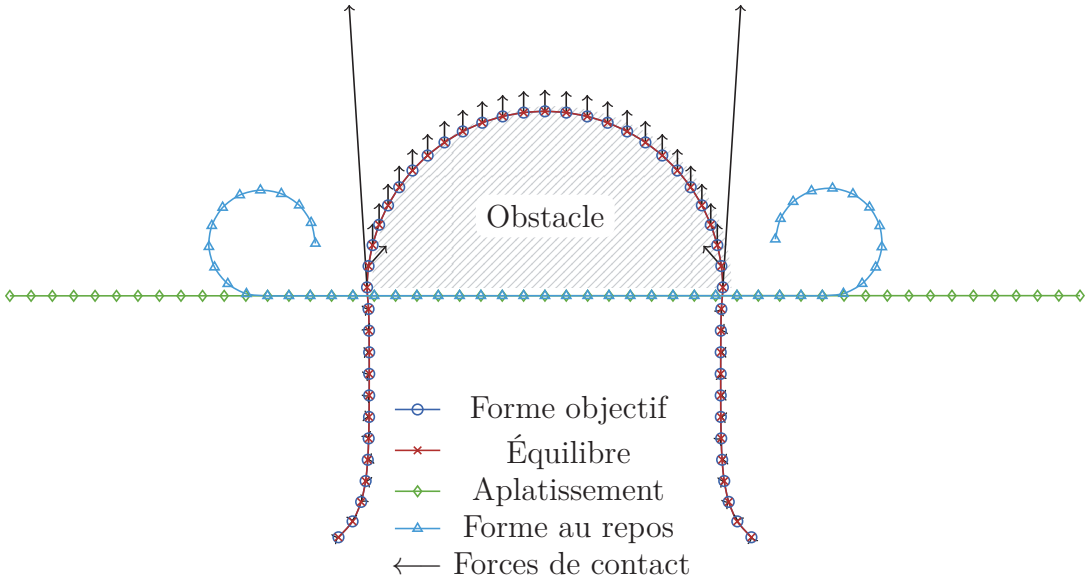
[2] *Free Form Deformation* : déformation en utilisant une « cage » englobante, facilement déformable.

[3] Encore merci à Laurence.

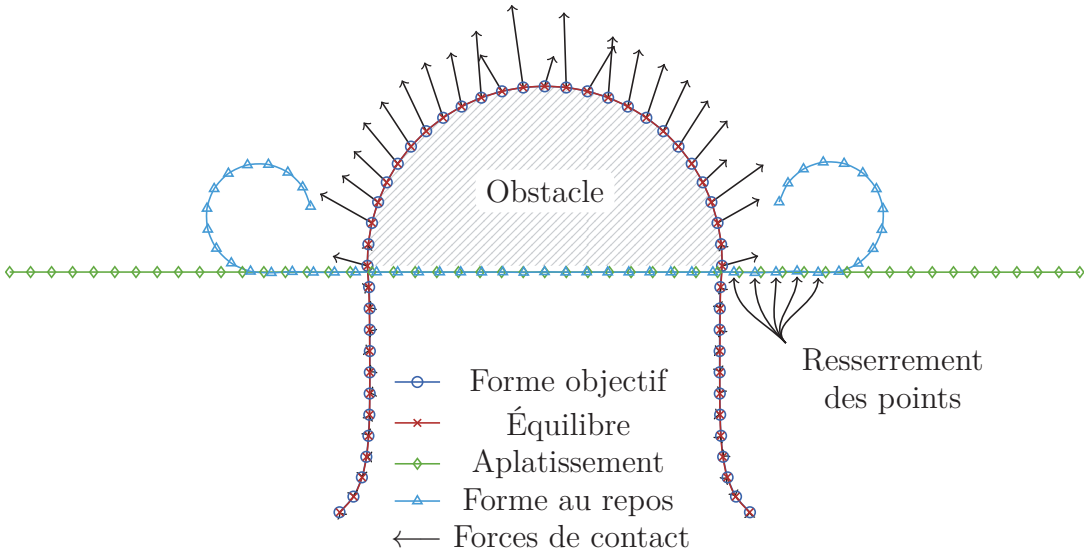
la taille de la région de stabilité n'est en revanche pas contrôlée. Rappelons que cet exemple est particulièrement difficile et qu'il n'existe peut-être pas de solution sous la forme sous laquelle nous la cherchons. En effet, les fronces induisent probablement de fortes rigidités locales qui ne sont pas modélisées (nous utilisons des raideurs constantes sur tout le maillage).

L'exemple de la jupe est quant à lui particulièrement bien traité par notre algorithme. Il parvient en effet à resserrer la jupe autour de la taille afin de la faire tenir sur le personnage grâce au frottement. La figure 11.2.4$_{147}$ permet de bien se rendre compte de ceci. De plus, la simulation d'un fort vent soufflant sur la jupe, figure 11.2.5$_{147}$, montre une excellente stabilité.

À propos du temps de calcul, même si la rapidité n'était pas l'objectif premier – trouver une manière d'attaquer le problème était déjà difficile –, nous sommes conscients qu'un temps d'exécution trop long rendrait la méthode peu utilisable en pratique. Le modèle le plus long à traiter a été la jupe et ses 5208 points pour laquelle le calcul a duré un peu moins de 3h. L'exemple de la nappe a été résolu en moins de 2min et celui de la manche ballon en un peu plus de 10min.



(a) Inversion avec fixation des points.



(b) Inversion avec contact unilatéral et frottement sec ($\mu = 0{,}2$).

FIGURE 11.1.1 : Comparaison de l'inversion avec notre approche en utilisant (a) le contact bilatéral (fixation des points) et (b) le contact unilatéral avec frottement (de Coulomb). Les deux algorithmes sont initialisés avec une version aplatie de la forme objectif. Dans les deux cas, la forme au repos calculée donne un équilibre qui correspond parfaitement à la forme objectif. Figure (a), remarquer comme certaines forces sont dirigées vers l'obstacle ; d'autres forment un angle important avec la normale à la surface de contact, ce qui suggère un fort coefficient de frottement. Les points qui sont en contact sur la forme objectif ne sont en fait pas déplacés par l'algorithme d'inversion sur la forme au repos. En effet, ils sont à des positions identiques sur la forme de départ (aplatie). Figure (b), la forme au repos est avantageusement resserrée (regarder la différence avec la forme aplatie au centre) pour que les forces de contact soient à l'intérieur du cône de Coulomb.

FIGURE 11.2.1 : Simulation après inversion « naïve » où la forme objectif (à gauche) est utilisée comme forme au repos. À droite, la jupe glisse le long de la taille puis se stabilise, grâce au frottement, dans une position éloignée de la forme objectif (*c.f.* figure 11.2.2$_{146}$). Pour inverser cette forme, notre algorithme va resserrer automatiquement la juppe à la taille, *c.f.* figure 11.2.4$_{147}$. Seul le contact avec frottement de Coulomb est utilisé ici (aucun point n'est fixé).

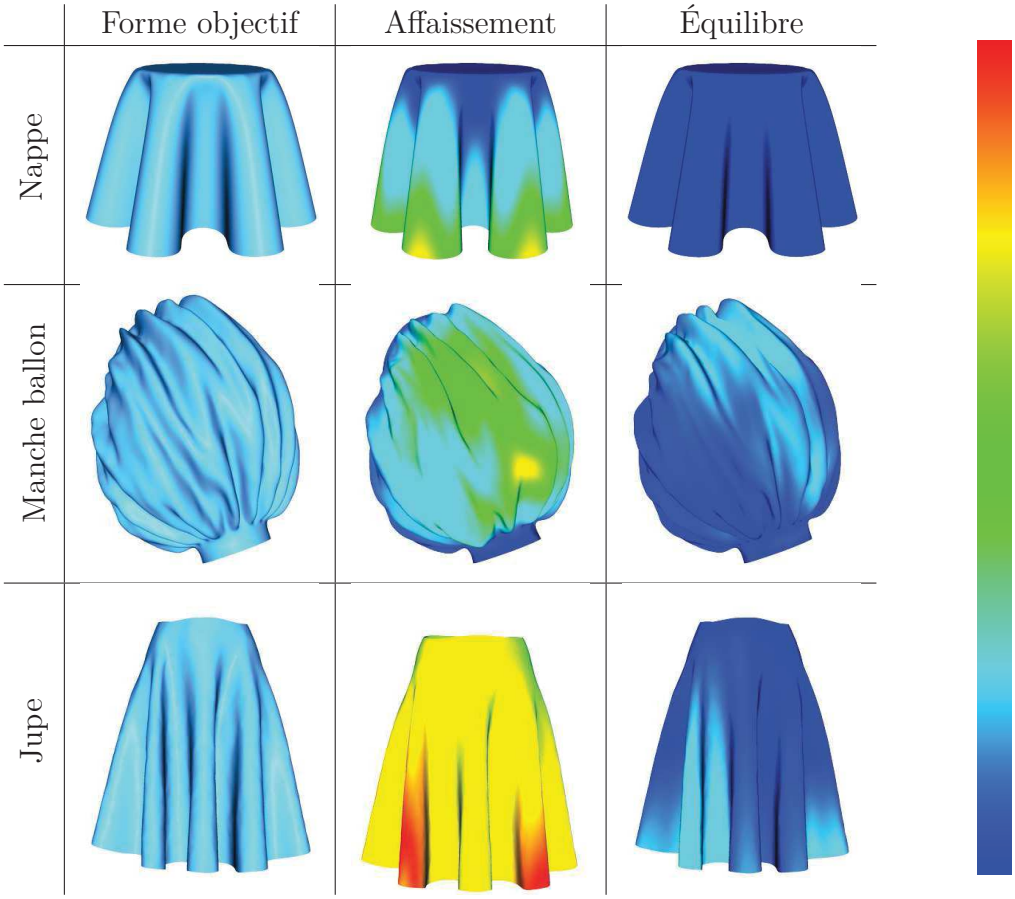


FIGURE 11.2.2 : Évaluation de notre méthode d'inversion sur trois exemples. De gauche à droite : la forme à inverser, l'équilibre obtenu par « inversion naïve » ($\bar{x} = x_{\mathrm{t}}$) et l'équilibre donné par notre méthode. Les formes des deux colonnes de droite sont colorées en fonction de leur distance à la forme objectif : la couleur bleu indique une distance nulle et le rouge une erreur de l'ordre de 10% d'une longueur typique de la scène.

FIGURE 11.2.3 : L'équilibre trouvé pour la manche ballon n'est pas très stable : un léger champ de force fait se « dégonfler » la manche.

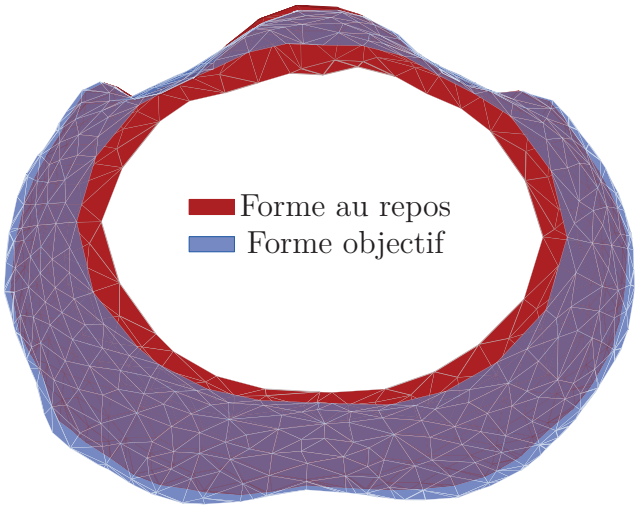


FIGURE 11.2.4 : Résultat de l'algorithme d'inversion appliqué à la jupe. On visualise ici, de dessus, la région de la taille (le reste de la jupe a été tronqué). Remarquer comment la forme au repos est resserrée par rapport à la forme objectif, de manière à faire tenir, grâce au frottement, la jupe sur la taille.

FIGURE 11.2.5 : Inversion puis simulation du modèle de la jupe avec notre approche. À gauche, la forme à l'équilibre est très proche de la forme objectif. Elle est notamment stabilisée grâce à un resserrement de la forme au repos à la taille (*c.f.* figure 11.2.4). À droite, le mouvement simulé en présence de vent est stable et réaliste. Seul le contact avec frottement de Coulomb est utilisé ici (aucun point n'est fixé).

Chapitre 12

Conclusion de la seconde partie

Dans cette seconde partie, nous nous sommes intéressés à l'élaboration d'un algorithme capable de retrouver la géométrie au repos d'une coque en contact frottant, connaissant sa forme à l'équilibre et les paramètres physiques du matériau qui la compose. Un tel algorithme trouve en effet son intérêt lorsque l'on souhaite simuler un objet dont on a un représentant (numérisé) « à l'équilibre » mais dont on ne connaît pas la forme au repos. En infographie par exemple, ceci permet de modéliser une coque sous la gravité et en contact avec d'autres objets sans se préoccuper de la physique. L'algorithme interprète alors la forme et s'occupe de donner les bons ingrédients au simulateur pour que cette forme soit effectivement un équilibre physique.

Résumé des contributions

Le premier ingrédient essentiel que nous avons utilisé, bien connu des spécialistes de problèmes inverses, a été la méthode de l'adjoint. Celle-ci nous a en effet permis de calculer de manière exacte le gradient de notre problème de minimisation aux moindres carrés. Ensuite, la multiplicité des équilibres, donnant au problème direct son caractère mal posé, nous a conduit à « guider » la méthode en pénalisant les équilibres éloignés de la forme objectif. Enfin, une reformulation du contact frottant en un problème d'optimisation, ainsi que la modification de la méthode de l'adjoint pour tenir compte du contact, nous ont finalement permis d'aboutir à un algorithme permettant d'attaquer le problème posé. Les résultats que nous avons obtenus sont très encourageants et nous ont permis de résoudre des cas complexes où l'algorithme se comportait de manière intuitive (resserrement de la jupe à la taille).

Limitations

Nous avons jusqu'ici supposé que les paramètres physiques du matériau étaient connus. La principale motivation de ce choix était que l'utilisateur a souvent une bonne idée du

type de comportement qu'il souhaite que la forme adopte une fois inversée. Cependant, nous avons remarqué que la convergence de la méthode était assez sensible aux paramètres physiques, notamment dans des cas complexes comme celui de la manche ballon. Donner plus de liberté à l'algorithme de ce point de vue, en tolérant une variation autour des paramètres physiques pris en entrée, pourrait sûrement permettre de rendre la convergence de la méthode plus précise et plus robuste.

Se pose aussi la question de l'utilisation d'une séquence de maillages animés, pour en déduire forme au repos et paramètres physiques. Le problème est en un sens plus complexe car le nombre de paramètres à estimer est plus important mais l'information, si elle est de qualité, est elle aussi plus conséquente. Les progrès récents de la capture d'objets en mouvement peuvent laisser penser que des méthodes robustes d'estimations de paramètres par capture de séquences de maillages vont devenir nécessaires. Ceci nous amène à la limitation suivante.

Dans ces travaux, nous n'avons pas présenté de résultats d'inversion de maillages issus de la capture. Nous avons pourtant tenté d'inverser un modèle 3D de manche, tiré de (White *et al.*, 2007), il est reproduit figure 12.0.1. La première difficulté que nous

FIGURE 12.0.1 : Modèle 3D de manche issu de la capture, tiré de (White *et al.*, 2007).

avons rencontrée est que le « contexte de contact » (les objets en contact avec le modèle) sont souvent absents car non capturés. Il a alors fallu « caler à la main » un objet censé expliquer la forme du modèle sous la gravité. Ensuite, certains triangles (quelques-uns) étaient incompatibles avec la simulation de vêtements, nous les avons retirés. Une fois le modèle prêt pour l'inversion, il s'est avéré que notre algorithme ne trouvait pas de forme au repos donnant un équilibre plus proche de la forme objectif que l'algorithme naïf. Cette approche naïve consiste, rappelons-le, à prendre comme forme au repos, la forme objectif. Notons que, le nombre de points de contact étant important, l'équilibre donné par l'approche naïve est déjà proche de la forme objectif et il est sans doute difficile de faire mieux.

Travaux futurs

En plus d'adresser les limitations évoquées ci-dessus, il existe plusieurs suites envisageables de ces travaux. Le fait d'avoir considéré que l'objet en entrée était une coque nous a permis qu'il soit sous une forme directement utilisable par le simulateur. Ceci en ignorant ostensiblement que les modèles 3D utilisés étaient parfois confectionnés à partir de patrons plats « cousus ». Une extension idéale de nos travaux, permettant de simuler les vêtements comme tels, serait une méthode capable de prendre en compte, dans le processus d'inversion, la recherche des découpes en patrons 2D. Un point de départ pourrait être de rechercher ces patrons sur la forme au repos retrouvée par notre algorithme plutôt que sur la forme objectif (Brouet *et al.*, 2012). En effet, les « patches » sur la forme objectif ne sont pas isométriques aux patrons 2D, alors que la forme retrouvée par notre algorithme a plus de chance de l'être puisqu'elle est une version non déformée du vêtement. Cette approche pourrait alors enrichir les méthodes de confection de patrons comme (Umetani *et al.*, 2011).

Chapitre 13

Perspectives

Même si les deux parties de cette thèse peuvent paraître distinctes, elles s'éclairent mutuellement sur plusieurs points et laissent ainsi envisager quelques perspectives.

Au cours de la seconde partie, consacrée à l'inversion de coques élastiques, nous nous sommes intéressés aux bonnes propriétés que devrait posséder un modèle élastique discret pour que son inversion soit facilitée. Il est apparu que notre modèle numérique de tiges élastiques de Kirchhoff, les Super-Clothoïdes 3D, élaboré dans la première partie, possédait toutes ces propriétés. Cela n'est pas étonnant quand on se souvient que ces caractéristiques sont partagées par le modèle des Super-Hélices et qu'elles ont permis, avec succès, l'inversion de chevelures complètes en présence de contact frottant (Derouet-Jourdan *et al.*, 2013a). Il apparaît donc intéressant d'envisager l'utilisation du modèle des Super-Clothoïdes 3D pour ce genre d'application. Ceci permettrait en effet d'utiliser une primitive plus riche et d'obtenir une meilleure précision pour un même nombre de degrés de liberté. Finalement, ce qui nous empêche pour l'instant d'appliquer cette méthode est un problème de conversion géométrique. En effet, on ne sait pas (encore) trouver la courbe en Clothoïde 3D par morceaux qui approche au mieux une courbe donnée par une suite de points. Dans le cas des Super-Hélices, cette conversion a d'ailleurs été une contribution cruciale (Derouet-Jourdan *et al.*, 2013b) pour le succès de l'inversion (Derouet-Jourdan *et al.*, 2013a). En dimension 2, la conversion d'une suite de points en clothoïdes par morceaux peut être considérée comme résolue (McCrae et Singh, 2008; Baran *et al.*, 2010) et a été exploitée pour l'inversion par Bertails-Descoubes (2012). Ajoutons, puisque nous avons déjà évoqué le cas des courbures et torsion polynomiales comme perspective, que la conversion d'une suite de points en une courbe dont les courbures et la torsion (matérielles) sont polynomiales par morceaux, serait une belle contribution géométrique.

La mise en évidence des qualités pertinentes pour qu'un modèle de coques élastiques (section 9.1_{100}) soit aisément inversible, nous a aussi permis de donner des pistes de réflexion pour l'élaboration d'un tel modèle, qui n'a, à notre avis, pas encore été proposé.

Ces pistes sont en partie inspirées des équations (géométriques) des tiges rencontrées dans la première partie ; elles permettent d'aboutir à une expression de l'énergie de flexion de coques similaire à celle utilisée dans le cas des tiges élastiques. Cependant, la difficulté de conception d'un tel modèle, par rapport aux tiges, tient selon nous en deux points. D'une part, contrairement au problème de Darboux dans le cas des tiges, les équations géométriques des surfaces sont assorties de conditions de compatibilité, dites de *Gauss* et *Codazzi–Mainardi*, ce qui rend leur intégration plus complexe que le problème de Cauchy de la géométrie d'une tige de Kirchhoff. D'autre part, le raccordement d'éléments (primitives) est bien plus difficile qu'en une seule dimension d'espace. En effet, les conditions de raccordement sont dans ce dernier cas soit de simples conditions initiales (continuité de la courbe et du repère) ou contraignent aisément le vecteur de courbures (continuité). Dans le cas des surfaces, ces conditions prennent la forme de conditions sur *tout*[1] le bord de l'élément ; ce bord étant lui-même défini par une courbe (fermée), plus ou moins complexe, dans l'espace des paramètres. Le problème de l'élaboration d'un tel modèle est donc difficile à notre avis et les pistes que nous avons proposées sont loin d'être suffisantes. Puisse l'avenir nous apporter une solution élégante à ce problème.

[1]Sauf dans le cas d'un élément qui aurait une portion de son bord libre, *i.e.*, raccordée à aucun autre élément.

Annexes

Annexe A

Forme explicite et forme fermée

À plusieurs reprises dans ce document, nous utilisons les termes « forme explicite » et « forme fermée » dont la distinction est importante pour comprendre notre travail. Nous tentons d'éclairer le lecteur à ce sujet dans cette annexe.

On dit que l'on connaît une *forme explicite* de la solution d'un problème, lorsque l'on peut en donner une expression mathématique. Cette terminologie s'oppose au fait que cette solution est définie *implicitement* par le problème. Par exemple, la solution de l'équation différentielle $y' = y$ assortie de la condition initiale $y(0) = 1$ possède une solution explicite $y(x) = e^x$. En revanche, on ne connaît pas de solution explicite au problème de Darboux $4.1.13_{33}$ dans le cas général. Pour le numéricien, posséder une forme explicite n'est pas forcément significatif. Parfois, une telle forme explicite permet un calcul stable, précis et rapide de la solution, parfois elle est inenvisageable numériquement et il est dans certains cas plus fructueux d'attaquer le problème directement. Par exemple, il existe une forme explicite au problème de Darboux (à droite) lorsque le vecteur de courbures est affine, elle est donnée[1] dans (Ivanova, 2001) ; cependant, ces formules font intervenir des expressions complexes à base d'intégrales et de la fonction hypergéométrique dégénérée dont le calcul direct semble difficilement envisageable numériquement. Finalement, ces formes explicites servent surtout à donner des résultats théoriques puisque leur manipulation peut être plus pratique que via le problème dont elles sont solutions.

On dit que la solution d'un problème est de *forme fermée* lorsqu'on dispose d'une forme explicite de cette solution qui s'exprime avec un nombre fini d'opérations et de fonctions « élémentaires ». L'ensemble de ces fonctions élémentaires dépend de l'application. Pour nous, une fonction sera élémentaire si elle est utilisable dans les langages de programmation classiques et que son implémentation est efficace (*i.e.*, ne requiert que très peu de cycles CPU[2]). Les fonctions exp, ln, cos ou sin en sont des exemples. Sous ces conditions, posséder une solution de forme fermée est beaucoup plus intéressant pour le numéricien : ceci garantit que l'on dispose d'une solution calculable par un ordina-

[1] En fait il resterait encore à expliciter la courbe moyenne en intégrant ces expressions.
[2] Ou FPU.

teur. Mais ceci ne promet pas nécessairement des calculs numériques efficaces, stables ou même précis ; un exemple simple est peut-être le calcul des solutions d'une équation du second degré lorsque $b^2 \gg 4ac$ (Higham, 2002, section 1.8 p. 10).

La solution à laquelle nous aboutissons pour le calcul de la solution du problème de Darboux est finalement très proche d'une expression en forme fermée sans vraiment pouvoir être désignée comme telle. En effet, d'une part les opérations nécessaires au calcul de ces séries sont effectivement élémentaires, ce ne sont que des opérations algébriques, d'autre part, les séries entières convergent si vite qu'une sommation finie est suffisante.

Annexe B

Rappels sur les séries entières

Définition B.0.1 (Série entière)**.** Une *série entière* est une série de fonctions de la forme $z \longmapsto \sum_{n\in\mathbb{N}} a_n z^n$ où z est une variable complexe et où $(a_n)_{n\in\mathbb{N}}$ est une suite complexe.

On peut s'interroger sur le domaine de convergence d'une série entière. On définit pour cela le *rayon de convergence* de la série entière qui est motivé par le lemme d'Abel.

Propriété B.0.2 (Lemme d'Abel)**.** *Soit $\sum_{n\in\mathbb{N}} a_n z^n$ une série entière et $z_0 \in \mathbb{C}^*$ tel que la suite $(a_n z_0^n)_{n\in\mathbb{N}}$ soit bornée. Alors*

(i) $\forall z \in \mathbb{C}$ tel que $|z| < |z_0|$, la série $\sum_{n\in\mathbb{N}} a_n z^n$ est absolument convergente ;

(ii) pour tout r tel que $0 < r < |z_0|$, la série de fonctions $\sum_{n\in\mathbb{N}} a_n z^n$ est normalement convergente sur $\{z \in \mathbb{C} \,/\, |z| \leqslant r\}$.

---

dem. Si M est un majorant de $(|a_n z_0^n|)_{n\in\mathbb{N}}$, la preuve est simple à partir de la majoration[1],

$$|a_n z^n| = \left|\frac{z}{z_0}\right|^n |a_n||z_0|^n \leqslant M \left|\frac{z}{z_0}\right|^n . \tag{B.0.3}$$

□

Ce lemme justifie la définition de rayon de convergence suivante.

Définition B.0.4 (Rayon de convergence)**.** Soit $\sum_{n\in\mathbb{N}} a_n z^n$ une série entière. Le nombre

$$R = \sup\{r \geqslant 0 \,/\, \text{la suite } (|a_n| r^n)_{n\in\mathbb{N}} \text{ est bornée}\} \tag{B.0.5}$$

est appelé *rayon de convergence* de la série entière $\sum_{n\in\mathbb{N}} a_n z^n$. D'après le lemme d'Abel,

- pour tout $z \in \mathbb{C}$ tel que $|z| < R$, $\sum_{n\in\mathbb{N}} a_n z^n$ converge absolument ;

[1] Remarquons que cette majoration est la même que celle de l'équation $4.2.7_{42}$ de la démonstration du théorème $4.2.1_{40}$.

- pour tout $z \in \mathbb{C}$ tel que $|z| > R$, $\sum_{n\in\mathbb{N}} a_n z^n$ diverge ;
- pour tout r tel que $0 \leqslant r < R$, la série entière $\sum_{n\in\mathbb{N}} a_n z^n$ converge normalement sur $\{z \in \mathbb{C} \,/\, |z| \leqslant r\}$.

Tiré de (Gourdon, 2009).

Annexe C

Théorème des fonctions implicites

Théorème C.0.1. *Soit U un ouvert de $\mathbb{R}^n \times \mathbb{R}^p$, (a, b) un point de U, et $f : (x, y) \longmapsto f(x, y)$ une application de classe $\mathscr{C}^1$ de U dans $\mathbb{R}^p$. On suppose que $f(a, b) = 0$ et que la différentielle $D_y f(a, b)$ est inversible. Alors l'équation $f(x, y) = 0$ peut être résolue localement par rapport à la variable y : il existe $V \subset \mathbb{R}^n$ un voisinage ouvert de a, $W \subset \mathbb{R}^p$ un voisinage ouvert de b, avec $V \times W \subset U$, et une application $\varphi : V \longrightarrow W$, de classe $\mathscr{C}^1$, unique, telle que*

$$(x \in V,\ y \in W \text{ et } f(x, y) = 0) \implies (x \in V \text{ et } y = \varphi(x)). \tag{C.0.2}$$

De plus, $D_y f(x, y)$ est inversible pour tout $(x, y) \in V \times W$, ce qui permet le calcul de $D\varphi(x)$ par la formule suivante,

$$D\varphi(x) = -\left[D_y f(x, \varphi(x))\right]^{-1} \circ D_x f(x, \varphi(x)), \tag{C.0.3}$$

pour tout x dans V.

Tiré de (Rouvière, 2009).

Annexe D

Quelques formules et résultats sur les rotations

D.1 Matrice de produit vectoriel

Pour $u = (u_1, u_2, u_3) \in \mathbb{R}^3$, on définit

$$[u]_\times = \begin{pmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{pmatrix}. \tag{D.1.1}$$

C'est la matrice du produit vectoriel à gauche par u puisqu'elle vérifie

$$\forall v \in \mathbb{R}^3, \quad [u]_\times v = u \times v. \tag{D.1.2}$$

Théorème D.1.3. *L'opérateur $[\,\cdot\,]_\times$ réalise un isomorphisme (d'espace vectoriel) de $\mathbb{R}^3$ dans $\mathcal{A}_3(\mathbb{R})$.*

---

dem. $[\,\cdot\,]_\times$ est linéaire puisque $[u+\lambda v]_\times w = (u+\lambda v)\times w = u \times w + \lambda v \times w = [u]_\times w + \lambda [v]_\times w$ d'où $[u+\lambda v]_\times = [u]_\times + \lambda [v]_\times$. De plus, $[\,\cdot\,]_\times$ est clairement injective puisque $[u]_\times = 0 \implies u = 0$. Enfin, puisque $\dim(\mathbb{R}^3) = \dim(\mathcal{A}_3(\mathbb{R}))$, $[\,\cdot\,]_\times$ est surjective, c'est donc bien une bijection. □

La plupart des identités sur le produit vectoriel ont un équivalent matriciel. Quelques-uns sont donnés dans le tableau suivant.

Version vectorielle	Version matricielle	Commentaire	
$(u \times v) \times w = (u \cdot w)v - (v \cdot w)u$	$[u \times v]_\times = vu^\top - uv^\top$	Double produit vectoriel	(I1)
$u \times (u \times v) = (u \cdot v)u - \|u\|^2 v$	${[u]_\times}^2 = uu^\top - \|u\|^2 I$		(I2)
$u \times (v \times w) + w \times (u \times v) + v \times (w \times u) = 0$	$[u \times v]_\times = [u]_\times [v]_\times - [v]_\times [u]_\times$	Identité de Jacobi, Crochet de Lie	(I3)
$\mathcal{R}(u \times v) = (\mathcal{R}u) \times (\mathcal{R}v)$	$[\mathcal{R}u]_\times = \mathcal{R}[u]_\times \mathcal{R}^\top$	$\mathcal{R} \in \mathcal{SO}(3)$	(I4)
$\det(u, v, w) = (u \times v) \cdot w$	$\det(u, v, w) = u^\top [v]_\times w$		(I5)

D.2 Rotations

Théorème D.2.1. *Le sous-espace stable par une rotation de $\mathcal{SO}(3)$ est de dimension 1 (axe de rotation) ou 3 (identité).*

---

dem. On cherche ici à montrer que pour $\mathcal{R} \in \mathcal{SO}(3)$, $\dim(\mathrm{Ker}(\mathcal{R} - I)) = 1$ ou 3. Il suffit pour cela de montrer que ce n'est ni 0 ni 2.

$$\det(\mathcal{R} - I) = \det\Big(\mathcal{R}^\top (I - \mathcal{R})\Big) = \det\Big(\mathcal{R}^\top\Big) \det(I - \mathcal{R}) = -\det(\mathcal{R} - I) = 0. \tag{D.2.2}$$

$\mathcal{R} - I$ n'étant pas inversible, la dimension de son noyau est au moins 1. Supposons que $\dim(\mathrm{Ker}(\mathcal{R} - I)) \geqslant 2$; il existe donc deux vecteurs non colinéaires u_1 et u_2 qui soient stables par $\mathcal{R}$. Ils vérifient

$$\mathcal{R}(u_1 \times u_2) = (\mathcal{R}u_1) \times (\mathcal{R}u_2) = u_1 \times u_2. \tag{D.2.3}$$

u_1, u_2 et $u_1 \times u_2$ sont donc trois vecteurs qui forment une famille libre de $\mathrm{Ker}(\mathcal{R} - I)$, il est donc de dimension 3. Finalement $\mathrm{Ker}(\mathcal{R} - I)$ ne peut pas être de dimension 2. □

Théorème D.2.4 (Formule de Rodrigues)**.** *Si $\theta \in \mathbb{R}$ et $n \in \mathcal{S}^2(\mathbb{R})$ alors*

$$\exp [\theta n]_\times = \sum_{k=0}^{+\infty} \frac{[\theta n]_\times{}^k}{k!} = \cos(\theta)\, I + \sin(\theta)\, [n]_\times + (1 - \cos(\theta))\, nn^\top \tag{D.2.5}$$

et c'est la rotation d'angle θ et d'axe engendré par n.

---

dem. On note $\mathcal{R} = \exp [\theta n]_\times$. I2$_{163}$ nous donne directement que $[n]_\times{}^2 = nn^\top - I$ et par récurrence, on obtient aisément que $[n]_\times{}^{2(p+1)} = (-1)^p \left(nn^\top - I\right)$ et que $[n]_\times{}^{2p+1} = (-1)^p [n]_\times$ pour $p \in \mathbb{N}$. D'où

$$\mathcal{R} = \sum_{k=0}^{+\infty} \frac{\theta^k}{k!} [n]_\times{}^k = \sum_{p=0}^{+\infty} \frac{\theta^{2p}}{(2p)!} [n]_\times{}^{2p} + \sum_{p=0}^{+\infty} \frac{\theta^{2p+1}}{(2p+1)!} [n]_\times{}^{2p+1} \tag{D.2.6a}$$

$$= I + \sum_{p=1}^{+\infty} (-1)^{p+1} \frac{\theta^{2p}}{(2p)!} \left(nn^\top - I\right) + \sum_{p=0}^{+\infty} (-1)^p \frac{\theta^{2p+1}}{(2p+1)!} [n]_\times \tag{D.2.6b}$$

$$= I + (1 - \cos(\theta)) \left(nn^\top - I\right) + \sin(\theta)\, [n]_\times = \cos(\theta)\, I + \sin(\theta)\, [n]_\times + (1 - \cos(\theta))\, nn^\top. \tag{D.2.6c}$$

Montrons maintenant que $\mathcal{R}$ est la rotation d'angle θ et d'axe engendré par n. Soit e_1 et e_2 deux vecteurs de $\mathcal{S}^2(\mathbb{R})$ tels que (e_1, e_2, n) soit une base orthonormée directe. D'après la formule de Rodrigues,

$$\mathcal{R}e_1 = \cos(\theta)e_1 + \sin(\theta)e_2 \tag{D.2.7a}$$

$$\mathcal{R}e_2 = \cos(\theta)e_2 - \sin(\theta)e_1 \tag{D.2.7b}$$

$$\mathcal{R}n = \cos(\theta)n + (1 - \cos(\theta))n = n \tag{D.2.7c}$$

donc dans la base (e_1, e_2, n), la matrice de $\mathcal{R}$ est

$$\begin{pmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{D.2.8}$$

$\mathcal{R}$ est donc bien la rotation d'angle θ autour de n. □

Remarque D.2.9. On retrouve ici le fait que l'exponentielle d'une matrice antisymétrique est une matrice de rotation (en tout cas pour la dimension 3). En géométrie différentielle, ce résultat est simplement que l'algèbre de Lie associée à $\mathcal{SO}(3)$ est $\mathcal{A}_3(\mathbb{R})$.

D.3 Géométrie différentielle sur $\mathcal{SO}(3)$

Théorème D.3.1. *Soit I un intervalle de $\mathbb{R}$ et $\mathcal{R} : I \longrightarrow \mathcal{SO}(3)$ de classe $\mathscr{C}^1$ sur I. Alors il existe $\kappa : I \longrightarrow \mathbb{R}^3$ et $\Omega : I \longrightarrow \mathbb{R}^3$ continues telles que*

$$\mathcal{R}' = \mathcal{R}[\kappa]_\times = [\Omega]_\times \mathcal{R}. \tag{D.3.2}$$

De plus, κ et Ω vérifient

$$\Omega = \mathcal{R}\kappa. \tag{D.3.3}$$

---

dem. C'est un résultat classique de géométrie différentielle sur les groupes de Lie puisque $T_{\mathcal{R}}\mathcal{SO}(3) = \mathcal{R}\,T_I\mathcal{SO}(3) = T_I\mathcal{SO}(3)\,\mathcal{R}$ et $T_I\mathcal{SO}(3) = \mathcal{A}_3(\mathbb{R}) = \left\{[u]_\times \,/\, u \in \mathbb{R}^3\right\}$. Mais amusons-nous quand-même à faire des calculs. Puisque $\mathcal{R} \in \mathcal{SO}(3)$, $\mathcal{R}^\top\mathcal{R} = I$, que l'on dérive pour obtenir

$$\mathcal{R}'^\top\mathcal{R} + \mathcal{R}^\top\mathcal{R}' = 0. \tag{D.3.4}$$

C'est donc que $\mathcal{R}^\top\mathcal{R}' \in \mathcal{A}_3(\mathbb{R})$. Par le théorème D.1.3$_{163}$, il existe $\kappa \in \mathbb{R}^3$ tel que

$$\mathcal{R}^\top\mathcal{R}' = [\kappa]_\times. \tag{D.3.5}$$

On remarque ici que $[\kappa]_\times$, donc κ, est bien continue. On construit Ω de la même manière en dérivant $\mathcal{R}\mathcal{R}^\top = I$ ou en utilisant l'identité I4$_{163}$. □

Bibliographie

Vincent Acary et Bernard Brogliato : *Numerical Methods for Nonsmooth Dynamical Systems : Applications in Mechanics and Electronics*, volume 35 de *Lecture Notes in Applied and Computational Mechanics*. Springer Verlag, 2008. URL `https://hal.inria.fr/inria-00423530`. (Cité page 125.)

Stuart Antman : *Nonlinear Problems of Elasticity (Applied Mathematical Sciences)*. Springer, 2005. ISBN 0387208801. URL `http://link.springer.com/book/10.1007/0-387-27649-1`. (Cité page 103.)

Uri M. Ascher et Eddy Boxerman : On the modified conjugate gradient method in cloth simulation. *The Visual Computer*, 19(7-8):526–531, 2003. (Cité page 124.)

Basile Audoly et Yves Pomeau : *Elasticity and geometry : from hair curls to the nonlinear response of shells*. Oxford University Press, 2010. (Cité pages 23 et 32.)

David Baraff : Linear-time dynamics using lagrange multipliers. *In Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '96, pages 137–146, New York, NY, USA, 1996. ACM. ISBN 0-89791-746-4. (Cité page 25.)

David Baraff et Andrew Witkin : Large steps in cloth simulation. *In Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '98, pages 43–54, New York, NY, USA, 1998. ACM. ISBN 0-89791-999-8. (Cité pages 26, 95, 106, 107, 109, 124, 136 et 139.)

Ilya Baran, Jaakko Lehtinen et Jovan Popovic : Sketching clothoid splines using shortest paths. *Computer Graphics Forum*, 29(2):655–664, 2010. (Cité pages 27 et 153.)

James V. Beck et Keith A. Woodbury : Inverse problems and parameter estimation : integration of measurements and analysis. *Measurement Science and Technology*, 9 (6):839, 1998. URL `http://stacks.iop.org/0957-0233/9/i=6/a=001`. (Cité page 97.)

Craig J. Benham et Steven P. Mielke : DNA Mechanics. *Annual Review of Biomedical Engineering*, 7(1):21–53, 2005. (Cité page 23.)

Miklós Bergou, Basile Audoly, Etienne Vouga, Max Wardetzky et Eitan Grinspun : Discrete Viscous Threads. *SIGGRAPH (ACM Transactions on Graphics)*, 29 (4), Jul 2010. (Cité pages 25, 74 et 75.)

Miklós Bergou, Max Wardetzky, Stephen Robinson, Basile Audoly et Eitan Grinspun : Discrete Elastic Rods. *ACM Transactions on Graphics (SIGGRAPH)*, 27(3):63 :1–63 :12, aug 2008. (Cité pages 25, 28, 29, 40, 73 et 74.)

Florence Bertails : *Simulation de Chevelures Virtuelles*. Thèse de doctorat, Institut National Polytechnique de Grenoble, 2006. URL `http://www-evasion.imag.fr/Publications/2006/Ber06`. (Cité page 37.)

Florence Bertails, Basile Audoly, Marie-Paule Cani, Bernard Querleux, Frédéric Leroy et Jean-Luc Lévêque : Super-helices for predicting the dynamics of natural hair. *In ACM SIGGRAPH 2006 Papers*, SIGGRAPH '06, pages 1180–1187, New York, NY, USA, 2006. ACM. ISBN 1-59593-364-6. URL `http://doi.acm.org/10.1145/1179352.1142012`. (Cité pages 19, 25, 28, 29, 35, 40, 44, 55, 63, 68, 73, 85 et 112.)

Florence Bertails-Descoubes : Super-Clothoids. *Computer Graphics Forum*, 31 (2):509–518, mai 2012. URL `https://hal.inria.fr/hal-00658996`. (Cité pages 20, 26, 27, 35, 72, 92 et 153.)

Kiran S. Bhat, Christopher D. Twigg, Jessica K. Hodgins, Pradeep K. Khosla, Zoran Popović et Steven M. Seitz : Estimating cloth simulation parameters from video. *In Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '03, pages 37–51, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association. ISBN 1-58113-659-5. URL `http://dl.acm.org/citation.cfm?id=846276.846282`. (Cité page 97.)

Alejandro Blumentals, Florence Bertails-Descoubes et Romain Casati : Dynamics of a developable shell with uniform curvatures. *In The 4th Joint International Conference on Multibody System Dynamics*, Montréal, Canada, mai 2016. URL `https://hal.inria.fr/hal-01311559`. (Cité page 113.)

Katherine L. Bouman, Bei Xiao, Peter Battaglia et William T. Freeman : Estimating the material properties of fabric from video. *In Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 1984–1991, Dec 2013. (Cité page 97.)

Derek Bradley, Tamy Boubekeur et Wolfgang Heidrich : Accurate multi-view reconstruction using robust binocular stereo and surface meshing. *In Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8, June 2008. (Cité pages 92 et 96.)

Robert Bridson, Ronald Fedkiw et John Anderson : Robust treatment of collisions, contact and friction for cloth animation. *ACM Transaction on Graphics*, 21(3):594–603, juillet 2002. ISSN 0730-0301. URL `http://doi.acm.org/10.1145/566654.566623`. (Cité page 95.)

Robert Bridson, Sebastian Marino et Ronald Fedkiw : Simulation of clothing with folds and wrinkles. *In Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '03, pages 28–36, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association. ISBN 1-58113-659-5. URL `http://dl.acm.org/citation.cfm?id=846276.846281`. (Cité pages 95 et 107.)

Remi Brouet, Alla Sheffer, Laurence Boissieux et Marie-Paule Cani : Design preserving garment transfer. *ACM Transaction on Graphics*, 31(4):36 :1–36 :11, juillet 2012. ISSN 0730-0301. URL `http://doi.acm.org/10.1145/2185520.2185532`. (Cité pages 96, 99 et 151.)

Florent Cadoux : *Optimisation et analyse convexe pour la dynamique non-régulière*. Theses, Université Joseph-Fourier - Grenoble I, novembre 2009. URL `https://tel.archives-ouvertes.fr/tel-00440798`. (Cité page 132.)

Peter B. Canham : The minimum energy of bending as a possible explanation of the biconcave shape of the human red blood cell. *Journal of Theoretical Biology*, 26 (1):61 – 81, 1970. ISSN 0022-5193. URL `http://www.sciencedirect.com/science/article/pii/S0022519370800327`. (Cité page 107.)

Michel Carignan, Ying Yang, Nadia Magnenat-Thalmann et Daniel Thalmann : Dressing animated synthetic actors with complex deformable clothes. *SIGGRAPH Computer Graphics*, 26(2):99–104, juillet 1992. ISSN 0097-8930. URL `http://doi.acm.org/10.1145/142920.134017`. (Cité page 96.)

Romain Casati et Florence Bertails-Descoubes : Super Space Clothoids. *ACM Transactions on Graphics*, 32(4):Article No. 48, juillet 2013. URL `http://bipop.inrialpes.fr/~casati/publications/ssc.html`. Special Issue : SIGGRAPH 2013 Conference. (Cité pages 11 et 30.)

Romain Casati, Gilles Daviet et Florence Bertails-Descoubes : Inverse Elastic Cloth Design with Contact and Friction. Research report, Inria Grenoble Rhône-Alpes, Université de Grenoble, avril 2016. URL `https://hal.archives-ouvertes.fr/hal-01309617`. (Cité page 116.)

CGSociety : SIGGRAPH's Animation Mother. `http://www.cgsociety.org/index.php/CGSFeatures/CGSFeatureSpecial/siggraph_animation_mother`, 2008. (Cité pages 81 et 82.)

Xiang Chen, Changxi Zheng, Weiwei Xu et Kun Zhou : An asymptotic numerical method for inverse elastic shape design. *ACM Transaction on Graphics*, 33(4):95 :1–95 :11, juillet 2014. ISSN 0730-0301. URL `http://doi.acm.org/10.1145/2601097.2601189`. (Non cité dans ce document.)

Nuttapong Chentanez, Ron Alterovitz, Daniel Ritchie, Lita Cho, Kris K. Hauser, Ken Goldberg, Jonathan R. Shewchuk et James F. O'Brien : Interactive

simulation of surgical needle insertion and steering. *In Proceedings of ACM SIGGRAPH 2009*, pages 88 :1–10, Aug 2009. URL `http://graphics.berkeley.edu/papers/Chentanez-ISN-2009-08/`. (Cité page 23.)

Philippe G. Ciarlet : *Mathematical elasticity. volume I. , Three-dimensional elasticity.* Studies in mathematics and its applications. Elsevier Science, Amsterdam, New York, 1988. ISBN 0-444-70259-8. (Non cité dans ce document.)

Philippe G. Ciarlet : *Mathematical elasticity. volume II. , Theory of plates*, volume 27 de *Studies in mathematics and its applications.* Elsevier Science, Amsterdam, Lausanne, New York, 1997. ISBN 0-444-82570-3. (Cité page 103.)

Philippe G. Ciarlet : *Mathematical elasticity. vol. 3 : theory of shells.* Studies in mathematics and its applications. Elsevier Science, Amsterdam, New York, Oxford, 2000. ISBN 0-444-82891-5. (Cité pages 101, 103 et 105.)

Philippe G. Ciarlet : *An Introduction to Differential Geometry with Applications to Elasticity.* Available online. Springer, 2006. ISBN 9781402042485. (Cité pages 101, 103 et 105.)

David Cohen-Steiner et Jean-Marie Morvan : Restricted delaunay triangulations and normal cycle. *In Proceedings of the Nineteenth Annual Symposium on Computational Geometry*, SCG '03, pages 312–321, New York, NY, USA, 2003. ACM. ISBN 1-58113-663-3. URL `http://doi.acm.org/10.1145/777792.777839`. (Cité page 106.)

François Cosserat et Eugène Cosserat : *Théorie des corps déformables.* A. Hermann et fils, 1909. (Cité pages 23 et 30.)

CreativeBloq : The making of pixar brave, 2012. URL `http://www.creativebloq.com/making-pixars-brave-8123080`. (Cité page 96.)

Michael A. Crisfield et Gordan Jelenić : Objectivity of strain measures in the geometrically exact three-dimensional beam theory and its finite-element implementation. *Proceedings of the Royal Society of London A : Mathematical, Physical and Engineering Sciences*, 455(1983):1125–1147, 1999. ISSN 1364-5021. (Cité page 25.)

Gaston Darboux : *Leçons sur la Théorie Générale des Surfaces.* Gauthier-Villars, 1887. URL `http://www.archive.org/details/leconsurlagendes04darbrich`. (Cité page 33.)

Gilles Daviet, Florence Bertails-Descoubes et Laurence Boissieux : A Hybrid Iterative Solver for Robustly Capturing Coulomb Friction in Hair Dynamics. *ACM Transactions on Graphics*, 30(6):139 :1–139 :12, 2011. URL `https://hal.inria.fr/hal-00647497`. (Cité pages 11, 26, 81, 91, 94, 98 et 125.)

Yves de Montaudouin, Wayne Tiller et Havard Vold : Applications of power series in computational geometry. *Computer-Aided Design*, 18(10):514 – 524,

1986. ISSN 0010-4485. URL http://www.sciencedirect.com/science/article/pii/0010448586900382. (Non cité dans ce document.)

Hervé Delingette, Martial Hebert et Katsushi Ikeuchi : Trajectory generation with curvature constraint based on energy minimization. *In Proceedings 1991 IEEE/RSJ International Conference On Intelligent Robotic Systems (IROS '91*, November 1991. (Cité page 26.)

Jean-Pierre Demailly : *Analyse numérique et équations différentielles*. Collection Grenoble sciences. EDP Sciences, 2006. ISBN 9782868838919. (Cité pages 34 et 64.)

Alexandre Derouet-Jourdan, Florence Bertails-Descoubes, Gilles Daviet et Joëlle Thollot : Inverse dynamic hair modeling with frictional contact. *ACM Transaction on Graphics*, 32(6):159 :1–159 :10, novembre 2013a. ISSN 0730-0301. URL https://hal.inria.fr/hal-00857559. (Cité pages 91, 98, 100, 101, 108, 125 et 153.)

Alexandre Derouet-Jourdan, Florence Bertails-Descoubes et Joëlle Thollot : Stable inverse dynamic curves. *ACM Transaction on Graphics*, 29(6):137 :1–137 :10, décembre 2010. ISSN 0730-0301. URL http://doi.acm.org/10.1145/1882261.1866159. (Cité pages 98 et 108.)

Alexandre Derouet-Jourdan, Florence Bertails-Descoubes et Joëlle Thollot : Floating tangents for approximating spatial curves with piecewise helices. *Computer Aided Geometric Design*, 30(5):490 – 520, 2013b. ISSN 0167-8396. URL http://www.sciencedirect.com/science/article/pii/S0167839613000289. (Cité pages 91, 101 et 153.)

Marcelo A. Dias et Basile Audoly : "Wunderlich, Meet Kirchhoff" : A general and unified description of elastic ribbons and thin rods. *Journal of Elasticity*, 119(1-2):49–66, 2015. ISSN 0374-3535. URL http://dx.doi.org/10.1007/s10659-014-9487-0. (Cité page 81.)

Ellis Harold Dill : Kirchhoff's theory of rods. *Archive for History of Exact Sciences*, 44(1):1–23, 1992. ISSN 0003-9519. (Cité pages 19 et 23.)

Franki J. E. Dillen : The classification of hypersurfaces of a euclidean space with parallel higher order fundamental form. *Math Z*, 203(1):635–643, jan 1990. (Cité page 27.)

Manfredo P. Do-Carmo : *Differential Geometry of Curves and Surfaces*. Prentice Hall, first édition, 1976. ISBN 0132125897. (Cité pages 103, 109 et 110.)

Elliot English et Robert Bridson : Animating developable surfaces using nonconforming elements. *ACM Transaction on Graphics*, 27(3):66 :1–66 :5, août 2008. ISSN 0730-0301. URL http://doi.acm.org/10.1145/1360612.1360665. (Cité pages 95 et 108.)

Thierry Fraichard et Alexis Scheuer : From reeds and shepp's to continuous-curvature paths. *Robotics, IEEE Transactions on*, 20(6):1025–1035, Dec 2004. ISSN 1552-3098. (Cité page 27.)

David Goldberg : What every computer scientist should know about floating-point arithmetic. *ACM Computing Surveys*, 23(1):5–48, mars 1991. ISSN 0360-0300. URL `http://doi.acm.org/10.1145/103162.103163`. (Cité page 46.)

Rony Goldenthal, David Harmon, Raanan Fattal, Michel Bercovier et Eitan Grinspun : Efficient Simulation of Inextensible Cloth. *SIGGRAPH (ACM Transactions on Graphics)*, 26(3), Jul 2007. (Cité pages 25, 95, 108 et 109.)

Herbert Goldstein, Charles P. Poole et John L. Safko : *Classical Mechanics (3rd Edition)*. Addison-Wesley, 3 édition, juin 2001. ISBN 0201657023. (Cité page 59.)

Alain Goriely et Sébastien Neukirch : Mechanics of climbing and attachment in twining plants. *Physical Review Letters*, 97:184302, Nov 2006. URL `http://www.lmm.jussieu.fr/~neukirch/publications/climbing_2d.html`. (Cité pages 23, 79 et 81.)

Xavier Gourdon : *Analyse (French Edition)*. Ellipses Marketing, 2009. ISBN 2729837590. (Cité page 160.)

Sachin Goyal, Noel C. Perkins et Christopher L. Lee : Non-linear dynamic intertwining of rods with self-contact. *International Journal of Non-Linear Mechanics*, 43 (1):65 – 73, 2008. ISSN 0020-7462. URL `http://www.sciencedirect.com/science/article/pii/S0020746207001989`. (Cité pages 23 et 24.)

Torbjörn Granlund et the GMP development team : *GNU MP : The GNU Multiple Precision Arithmetic Library*, 5.0.5 édition, 2012. `http://gmplib.org/`. (Cité page 50.)

Eitan Grinspun, Anil N. Hirani, Mathieu Desbrun et Peter Schröder : Discrete shells. *In Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '03, pages 62–67, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association. ISBN 1-58113-659-5. URL `http://dl.acm.org/citation.cfm?id=846276.846284`. (Cité pages 12, 95, 107 et 108.)

Eran Guendelman, Andrew Selle, Frank Losasso et Ronald Fedkiw : Coupling water and smoke to thin deformable and rigid shells. *ACM Transaction on Graphics*, 24(3):973–981, juillet 2005. ISSN 0730-0301. URL `http://doi.acm.org/10.1145/1073204.1073299`. (Cité page 107.)

Jacques Hadamard : Sur les problèmes aux dérivés partielles et leur signification physique. *Princeton University Bulletin*, 13:49–52, 1902. (Cité page 117.)

Sunil Hadap : Oriented strands : Dynamics of stiff multi-body system. *In Proceedings of the 2006 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '06, pages 91–100, Aire-la-Ville, Switzerland, Switzerland, 2006. Eurographics Association. ISBN 3-905673-34-7. (Cité pages 25 et 28.)

Sunil Hadap et Nadia Magnenat-Thalmann : Modeling dynamic hair as a continuum. *Computer Graphics Forum*, 20(e 3):329–338, September 2001. (Cité page 25.)

Fabian Hahn, Bernhard Thomaszewski, Stelian Coros, Robert W. Sumner, Forrester Cole, Mark Meyer, Tony DeRose et Markus Gross : Subspace clothing simulation using adaptive bases. *ACM Transaction on Graphics*, 33(4):105 :1–105 :9, juillet 2014. ISSN 0730-0301. URL `http://doi.acm.org/10.1145/2601097.2601160`. (Cité page 95.)

Ernst Hairer, Christian Lubich et Gerhard Wanner : *Geometric Numerical Integration : Structure-Preserving Algorithms for Ordinary Differential Equations (Springer Series in Computational Mathematics)*. Springer, 2006. ISBN 3540306633. URL `http://www.springer.com/cn/book/9783540306634`. (Cité pages 70 et 79.)

Gur Harary et Ayellet Tal : 3d euler spirals for 3d curve completion. *Computational Geometry*, 45(3):115 – 126, 2012. ISSN 0925-7721. (Cité pages 20 et 27.)

Nils Hasler, Bodo Rosenhahn et Hans-Peter Seidel : Reverse engineering garments. *In* André Gagalowicz et Wilfried Philips, éditeurs : *Computer Vision/Computer Graphics Collaboration Techniques*, volume 4418 de *Lecture Notes in Computer Science*, pages 200–211. Springer Berlin Heidelberg, 2007. ISBN 978-3-540-71456-9. URL `http://dx.doi.org/10.1007/978-3-540-71457-6_19`. (Cité page 97.)

Wolfgang Helfrich : Elastic properties of lipid bilayers : theory and possible experiments. *Zeitschrift für Naturforschung. Teil C : Biochemie, Biophysik, Biologie, Virologie*, 28(11):693–703, 1973. ISSN 0341-0471. URL `http://view.ncbi.nlm.nih.gov/pubmed/4273690`. (Cité page 107.)

Nicholas J. Higham : *Accuracy and Stability of Numerical Algorithms*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2nd édition, 2002. ISBN 0898715210. (Cité page 158.)

Alan C. Hindmarsh : ODEPACK, a systematized collection of ODE solvers. *IMACS Transactions on Scientific Computation*, 1:55–64, 1983. URL `http://www.netlib.org/odepack/`. (Cité page 70.)

Jean-Baptiste Hiriart-Urruty et Claude Lemaréchal : *Convex analysis and minimization algorithms. I. , Fundamentals*. Grundlehren der mathematischen Wissenschaften. Springer 1996, Berlin, Heidelberg, New York, 1993. ISBN 3-540-56850-6. Autre tirage : 2009. (Cité pages 127 et 133.)

Jean-Baptiste Hiriart-Urruty et Claude Lemaréchal : *Convex analysis and minimization algorithms. II. , Advanced theory and bundle methods*. Grundlehren der mathematischen Wissenschaften. Springer-Verlag, Berlin, New-york, 1996. ISBN 3-540-56852-2. (Non cité dans ce document.)

Takeo Igarashi et John F. Hughes : Clothing manipulation. *ACM Transaction on Graphics*, 22(3):697–697, juillet 2003. ISSN 0730-0301. URL `http://doi.acm.org/10.1145/882262.882328`. (Cité page 96.)

Elena A. Ivanova : On the use of a new method of solution of Darboux problem for solution of the problem of motion of a ball on a routh plane. *In XXVIII Summer School Nonlinear oscillations in Mechanical Systems. Actual Problems in Mechanics*, pages 361–367, 2001. URL `http://www.ipme.ru/ipme/labs/dms/prive/ivanova/Home_page_Elena_Ivanova/PDF/APM2000.pdf`. (Cité page 157.)

Nebojsa Jojic et Thomas S. Huang : Estimating cloth draping parameters from range data. *In In International Workshop on Synthetic-Natural Hybrid Coding and 3-D Imaging*, pages 73–76, 1997. (Cité page 97.)

Alonzo Kelly et Bryan Nagy : Reactive nonholonomic trajectory generation via parametric optimal control. *The International Journal of Robotics Research*, 22(7 - 8):583 – 601, July 2003. (Cité pages 27 et 72.)

Michel Kern : Problèmes inverses, polycopié du cours donné à l'École Supérieure d'Ingénieurs Léonard de Vinci, 2002. URL `https://who.rocq.inria.fr/Michel.Kern/Teaching/ESILV/inverse.pdf`. (Cité page 89.)

Doyub Kim, Woojong Koh, Rahul Narain, Kayvon Fatahalian, Adrien Treuille et James F. O'Brien : Near-exhaustive precomputation of secondary cloth effects. *ACM Transaction on Graphics*, 32(4):87 :1–87 :8, juillet 2013. ISSN 0730-0301. URL `http://doi.acm.org/10.1145/2461912.2462020`. (Cité page 95.)

Benjamin B. Kimia, Ilana Frankel et Ana-Maria Popescu : Euler spiral for shape completion. *International Journal of Computer Vision*, 54(1-3):159–182, 2003. ISSN 0920-5691. (Cité page 27.)

Raphael Linus Levien : *From Spiral to Spline : Optimal Techniques in Interactive Curve Design*. Thèse de doctorat, EECS Department, University of California, Berkeley, Dec 2009. URL `http://www.levien.com/phd/phd.html`. (Cité page 27.)

Yong-Jin Liu, Dong-Liang Zhang et Matthew Ming-Fai Yuen : A survey on {CAD} methods in 3d garment design. *Computers in Industry*, 61(6):576 – 593, 2010. ISSN 0166-3615. URL `http://www.sciencedirect.com/science/article/pii/S0166361510000242`. Soft Products Development. (Cité page 95.)

Marvelous Designer : Marvelous Designer., 2010. URL `http://www.marvelousdesigner.com/`. (Cité pages 92 et 96.)

James McCrae et Karan Singh : Sketching piecewise clothoid curves. *In Proceedings of the Fifth Eurographics Conference on Sketch-Based Interfaces and Modeling*, SBM'08, pages 1–8, Aire-la-Ville, Switzerland, Switzerland, 2008. Eurographics Association. ISBN 978-3-905674-07-1. (Cité pages 27 et 153.)

Matthew McNaughton, Christopher Urmson, John M Dolan et Jin-Woo Lee : Motion planning for autonomous driving with a conformal spatiotemporal lattice. *In Proceedings of the International Conference on Robotics and Automation*, pages 4889–4895, May 2011. (Cité pages 27 et 72.)

Marc Mezzarobba : *Autour de l'évaluation numérique des fonctions D-finies*. Theses, Ecole Polytechnique X, octobre 2011. URL `https://pastel.archives-ouvertes.fr/pastel-00663017`. (Cité page 20.)

Markus Neher : An enclosure method for linear ODEs with analytical coefficients. *ZAMM*, 76(S1):489–490, 1996. (Cité pages 41, 52 et 53.)

Markus Neher : An enclosure method for the solution of linear ODEs with polynomial coefficients. *Numerical Functional Analysis and Optimization*, 20:779–803, 1999. (Cité pages 41, 52, 53 et 54.)

Jorge Nocedal et Stephen Wright : *Numerical Optimization*. Springer Series in Operations Research and Financial Engineering. Springer New York, 2006. ISBN 9780387303031. (Cité page 122.)

Dinesh K. Pai : STRANDS : Interactive Simulation of Thin Solids using Cosserat Models. *Computer Graphics Forum*, 21:347–352, 2002. (Cité page 25.)

Cliff Pickover : The official tangled cable appreciation page, 2012. URL `http://sprott.physics.wisc.edu/pickover/pc/cablerc.html`. (Cité page 11.)

Robert Piessens, Elise deDoncker Kapenga, Christoph W. Überhuber et David Kahaner : *QUADPACK : A Subroutine Package for Automatic Integration*. Springer–Verlag, 1983. (Cité page 73.)

Serban D. Porumbescu, Brian Budge, Louis Feng et Kenneth I. Joy : Shell maps. *ACM Transaction on Graphics*, 24(3):626–633, juillet 2005. ISSN 0730-0301. URL `http://doi.acm.org/10.1145/1073204.1073239`. (Cité page 96.)

Xavier Provot : Deformation constraints in a mass-spring model to describe rigid cloth behavior. *In Graphics Interface*, pages 147–154, 1995. (Cité page 109.)

François Rouvière : *Petit guide de calcul différentiel : à l'usage de la licence et de l'agrégation*. Cassini, 3ième édition, 2009. ISBN 2842250699. (Cité page 161.)

Julius Sachs, Alfred W. Bennett et William T. Thiselton-Dyer : *Textbook of botany, morphological and physiological. By Julius Sachs. Tr. and annotated by Alfred W. Bennett, assisted by W. T. Thiselton Dyer.* Oxford,Clarendon Press,, 1875. URL `http://www.biodiversitylibrary.org/item/`

76605. http ://www.biodiversitylibrary.org/bibliography/32340 — First edition, 1868. (Cité page 24.)

Andrew Selle, Michael Lentine et Ronald Fedkiw : A mass spring model for hair simulation. *ACM Transaction on Graphics*, 27(3):64 :1–64 :11, août 2008. ISSN 0730-0301. (Non cité dans ce document.)

Juan Carlos Simo et Loc Vu-Quoc : A three-dimensional finite-strain rod model. part II : Computational aspects. *Computer Methods in Applied Mechanics and Engineering*, 58(1):79–116, octobre 1986. ISSN 0045-7825. (Cité page 24.)

Jonas Spillmann et Matthias Teschner : CoRdE : Cosserat Rod Elements for the Dynamic Simulation of One-dimensional Elastic Objects. *In Proceedings of the 2007 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '07, pages 63–72, Aire-la-Ville, Switzerland, Switzerland, 2007. Eurographics Association. ISBN 978-1-59593-624-0. (Cité page 25.)

Rasmus Tamstorf et Eitan Grinspun : Discrete bending forces and their jacobians. *Graphical Models*, 75(6):362 – 370, 2013. ISSN 1524-0703. URL `http://www.sciencedirect.com/science/article/pii/S1524070313000209`. (Cité page 107.)

Demetri Terzopoulos, John Platt, Alan Barr et Kurt Fleischer : Elastically deformable models. *SIGGRAPH Computer Graphics*, 21(4):205–214, Aug 1987. ISSN 0097-8930. (Cité page 106.)

Bernhard Thomaszewski, Simon Pabst et Wolfgang Strasser : Continuum-based strain limiting. *Computer Graphics Forum*, 28(2):569–576, 2009. ISSN 1467-8659. URL `http://dx.doi.org/10.1111/j.1467-8659.2009.01397.x`. (Cité pages 95 et 108.)

Emmanuel Turquin, Jamie Wither, Laurence Boissieux, Marie-Paule Cani et John F. Hughes : A sketch-based interface for clothing virtual characters. *IEEE Computer Graphics and Applications*, 27(1):72–81, janvier 2007. URL `https://hal.archives-ouvertes.fr/hal-00171414`. Special issue on sketch based modeling. (Cité pages 92 et 96.)

Christopher D. Twigg et Zoran Kačić-Alesić : Optimization for sag-free simulations. *In Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '11, pages 225–236, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0923-3. URL `http://doi.acm.org/10.1145/2019406.2019437`. (Cité page 98.)

Nobuyuki Umetani, Danny M. Kaufman, Takeo Igarashi et Eitan Grinspun : Sensitive couture for interactive garment editing and modeling. *ACM Transactions on Graphics (SIGGRAPH 2011)*, 30(4), 2011. (Cité pages 92, 96, 107 et 151.)

Pascal Volino, Frederic Cordier et Nadia Magnenat-Thalmann : From early virtual garment simulation to interactive fashion design. *Computer Aided Design*, 37 (6):593–608, mai 2005. ISSN 0010-4485. URL `http://dx.doi.org/10.1016/j.cad.2004.09.003`. (Cité pages 92 et 96.)

Pascal Volino et Nadia Magnenat-Thalmann : Stop-and-go cloth draping. *The Visual Computer*, 23(9-11):669–677, Jun 2007. ISSN 1432-2315. (Cité page 122.)

Huamin Wang, James F. O'Brien et Ravi Ramamoorthi : Data-driven elastic models for cloth : Modeling and measurement. *ACM Transaction on Graphics*, 30(4):71 :1–71 :12, juillet 2011. ISSN 0730-0301. URL `http://doi.acm.org/10.1145/2010324.1964966`. (Cité page 95.)

Yuanzhen Wang, Beibei Liu et Yiying Tong : Linear surface reconstruction from discrete fundamental forms on triangle meshes. *Computer Graphics Forum*, 31(8): 2277–2287, 2012. ISSN 1467-8659. (Cité page 108.)

Kelly Ward, Florence Bertails, Tae yong Kim, Stephen R. Marschner, Marie paule Cani et Ming C. Lin : A survey on hair modeling : styling, simulation, and rendering. *In IEEE Transaction on Visualization and Computer Graphics*, pages 213–234, 2006. (Cité page 23.)

Max Wardetzky, Miklós Bergou, David Harmon, Denis Zorin et Eitan Grinspun : Discrete quadratic curvature energies. *Computer Aided Geometric Design*, 24(8–9): 499 – 518, 2007. ISSN 0167-8396. URL `http://www.sciencedirect.com/science/article/pii/S0167839607000891`. Discrete Differential Geometry. (Cité pages 107 et 108.)

Ryan White, Keenan Crane et David Forsyth : Capturing and animating occluded cloth. *In ACM Transactions on Graphics (SIGGRAPH)*, 2007. (Cité pages 12, 96 et 150.)

Eduardo H. Zarantonello : Projections on convex sets in Hilbert space and spectral theory. *In Contributions to Nonlinear Functional Analysis*, pages 237–424. Academic Press, 1971. URL `http://www.ams.org/mathscinet-getitem?mr=52:9014`. (Cité page 133.)

Résumé

Il est intéressant d'observer qu'une grande partie des objets déformables qui nous entourent sont caractérisés par une forme élancée : soit filiforme, comme les cheveux, les plantes, les fils ; soit surfacique, comme le papier, les feuilles d'arbres, les vêtements ou la plupart des emballages. Simuler (numériquement) la mécanique de telles structures présente alors un intérêt certain : cela permet de prédire leur comportement dynamique, leur forme statique ou encore les efforts qu'elles subissent. Cependant, pour pouvoir réaliser correctement ces simulations, plusieurs problèmes se posent. Les modèles (mécaniques, numériques) utilisés doivent être adaptés aux phénomènes que l'on souhaite reproduire ; le modèle mécanique choisi doit pouvoir être traité numériquement ; enfin, il est nécessaire de connaître les paramètres du modèle qui permettront de reproduire l'instance du phénomène souhaitée. Dans cette thèse nous abordons ces trois points, dans le cadre de la simulation de structures élancées.

Dans la première partie, nous proposons un modèle discret de tiges de Kirchhoff dynamiques, de haut degré, basé sur des éléments en courbures et torsion affines par morceaux : les Super-Clothoïdes 3D. Cette discrétisation spatiale est calculée de manière précise grâce à une méthode dédiée, adaptée à l'arithmétique flottante, utilisant des développements en séries entières. L'utilisation des courbures et de la torsion comme degrés de liberté permet d'aboutir à un schéma d'intégration stable grâce à une implicitation, à moindres frais, des forces élastiques. Le modèle a été utilisé avec succès pour simuler la croissance de plantes grimpantes ou le mouvement d'une chevelure. Nos comparaisons avec deux modèles de référence de la littérature ont montré que pour des tiges bouclées, notre approche offre un meilleur compromis en termes de précision spatiale, de richesse de mouvements générés et d'efficacité en temps de calcul.

Dans la seconde partie, nous nous intéressons à l'élaboration d'un algorithme capable de retrouver la géométrie au repos (non déformée) d'une coque en contact frottant, connaissant sa forme à l'équilibre et les paramètres physiques du matériau qui la compose. Un tel algorithme trouve son intérêt lorsque l'on souhaite simuler un objet pour lequel on dispose d'une géométrie (numérisée) « à l'équilibre » mais dont on ne connaît pas la forme au repos. En informatique graphique, un exemple d'application est la modélisation de vêtements virtuels sous la gravité et en contact avec d'autres objets : simplement à partir de la forme objectif et d'un simulateur de vêtement, le but consiste à identifier automatiquement les paramètres du simulateur tels que la forme d'entrée corresponde à un équilibre mécanique stable. La formulation d'un tel problème inverse comme un problème aux moindres carrés nous permet de l'attaquer avec la méthode de l'adjoint. Cependant, la multiplicité des équilibres, donnant au problème direct son caractère mal posé, nous conduit à « guider » la méthode en pénalisant les équilibres éloignés de la forme objectif. On montre enfin qu'il est possible de considérer du contact et du frottement solide dans l'inversion, en reformulant le calcul d'équilibres en un problème d'optimisation sous contraintes coniques, et en adaptant la méthode de l'adjoint à ce cas non-régulier. Les résultats que nous avons obtenus sont très encourageants et nous ont permis de résoudre des cas complexes où l'algorithme se comportait de manière intuitive.

Abstract

It is interesting to observe that many of the deformable objects around us are characterized by a slender structure: either in one dimension, like hair, plants, strands, or in two dimensions, such as paper, the leaves of trees or clothes. Simulating the mechanical behavior of such structures numerically is useful to predict their static shape, their dynamics, or the stress they undergo. However, to perform these simulations, several problems need to be addressed. First, the model (mechanical, numerical) should be adapted to the phenomena which it is aimed at reproducing. Then, the chosen mechanical model should be discretized consistently. Finally, it is necessary to identify the parameters of the model in order to reproduce a specific instance of the phenomenon. In this thesis we shall discuss these three points, in the context of the simulation of slender structures.

In the first part, we propose a discrete dynamic Kirchhoff rod model of high degree, based on elements with piecewise affine curvature and twist: the Super-Space-Clothoids. This spatial discretization is computed accurately through a dedicated method, adapted to floating-point arithmetic, using power series expansions. The use of curvature and twist as degrees of freedom allows us to make elastic forces implicit in the integration scheme. The model has been used successfully to simulate the growth of climbing plants or hair motion. Our comparisons with two reference models have shown that in the case of curly rods, our approach offers the best trade-off in terms of spatial accuracy, richness of motion and computational efficiency.

In the second part, we focus on identifying the undeformed configuration of a shell in the presence of frictional contact forces, knowing its shape at equilibrium and the physical parameters of the material. Such a method is of utmost interest in Computer Graphics when, for example, a user often wishes to model a virtual garment under gravity and contact with other objects regardless of physics. The goal is then to interpret the shape and provide the right ingredients to the cloth simulator, so that the cloth is actually at equilibrium when matching the input shape. To tackle such an inverse problem, we propose a least squares formulation which can be optimized using the adjoint method. However, the multiplicity of equilibria, which makes our problem ill-posed, leads us to "guide" the optimization by penalizing shapes that are far from the target shape. Finally, we show how it is possible to consider frictional contact in the inversion process by reformulating the computation of equilibrium as an optimization problem subject to conical constraints. The adjoint method is also adjusted to this non-regular case. The results we obtain are very encouraging and have allowed us to solve complex cases where the algorithm behaves intuitively.