



HAL
open science

Courbes, cylindres et métamorphoses pour l'image de synthèse

Francis Lazarus

► **To cite this version:**

Francis Lazarus. Courbes, cylindres et métamorphoses pour l'image de synthèse. Modélisation et simulation. Université Paris-Diderot - Paris VII, 1995. Français. NNT: . tel-00006284

HAL Id: tel-00006284

<https://theses.hal.science/tel-00006284>

Submitted on 17 Jun 2004

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

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

THÈSE

présentée à

L'UNIVERSITÉ DE PARIS VI

pour obtenir le titre de

DOCTEUR ES SCIENCES

Spécialité :

INFORMATIQUE

par

Francis LAZARUS

Sujet de la thèse :

COURBES, CYLINDRES ET MÉTAMORPHOSES
pour l'image de synthèse

Soutenue le 21 Décembre 1995, devant le jury composé de :

M.	Christian	CHOFFRUT	Président
M.	Claude	PUECH	Rapporteur
M.	Francis	SCHMITT	Rapporteur
M.	Pascal	LIENHARDT	Examineur
M.	André	GAGALOWICZ	Directeur de thèse
Mm.	Anne	VERROUST	Responsable de thèse

A Pascale et Mehdi

La forme belle, qu'elle soit un arbre ou un fleuve, les seins d'une femme ou ses flancs, les épaules ou les bras d'un homme ou le crâne d'un dieu, la forme belle c'est la forme qui s'adapte à sa fonction. L'idée n'a pas d'autre rôle que de nous la définir. L'idée, c'est l'aspect supérieur et l'extension infinie dans le monde et l'avenir du plus impérieux de nos instincts qu'elle résume et dénonce comme la fleur et le fruit résument la plante la prolongent et la perpétuent.

ELIE FAURE.

Remerciements

Ce travail a été réalisé dans d'excellentes conditions et je tiens à exprimer mes remerciements envers toutes celles et tous ceux qui ont contribué à son accomplissement :

Je remercie Anne Verroust qui par son encadrement et son soutien quotidien m'a guidé tout au long de cette thèse. Nos nombreuses discussions évoquent pour moi un idéal du monde de la recherche où l'écoute de l'autre et l'ouverture d'esprit s'accordent au son d'un travail harmonieux. Je la remercie également pour son aide précieuse dans la rédaction de ce manuscrit, je n'aurai pu espérer meilleur conseiller.

Je remercie André Gagalowicz qui m'a accueilli dans son laboratoire et a mis à ma disposition des moyens informatiques et audiovisuels à la mesure de ce qu'exige la synthèse d'images.

Je remercie Claude Puech et Francis Schmitt qui ont accepté d'être rapporteurs de cette thèse et de siéger à son jury. Qu'ils trouvent ici l'expression de ma gratitude pour l'intérêt qu'ils ont porté à mon travail.

Je remercie Christian Choffrut et Pascal Lienhardt pour l'intérêt qu'ils ont porté à mon travail et pour m'avoir fait l'honneur de bien vouloir faire partie de mon jury.

Je remercie l'équipe audiovisuelle de l'INRIA et tout particulièrement Arghyro Paouri, Christian Blonz et Alexandre Eidelman pour leur précieuse collaboration, pour leurs conseils et leur gentillesse.

Je remercie Jean-Paul Chieze qui m'a guidé dans les méandres de l'univers informatique toujours avec patience et gentillesse. Son importance est capitale dans la réalisation de ce travail.

Je remercie Philippe Decaudin, Fabrice Neyret et Frederic Taillefer, mes compagnons de bureau, pour l'aide qu'ils m'ont apportée et leur gentillesse à mon égard. Ils ont grandement

contribué à rendre ce travail agréable.

Un grand merci à Laurence Bourcier, notre secrétaire pour sa grande disponibilité et son extraordinaire patience, ainsi qu'à tous les membres du projet SYNTIM Bruno, Jean-philippe, Xavier, Christophe, Marie-Luce, Pierre, Sabine et tous les autres...

Je remercie Pascale pour sa lecture attentive de ce manuscrit, ainsi bien sûr que pour son soutien, sa confiance et l'aide inestimable qu'elle m'a apportée durant ces trois années de thèse.

Je remercie Mehdi dont la naissance a coïncidé avec le début de la rédaction de ce manuscrit et qui m'a fourni toute l'énergie nécessaire pour cette tâche difficile.

Enfin, je remercie tous mes proches pour leurs encouragements.

Table des matières

Introduction	1
I Interpolation de courbes	5
1 Un procédé canonique d'interpolation de courbes sur les groupes de Lie	9
1.1 Interpolation de points	9
1.2 Définition d'une courbe	10
1.3 Position du problème	11
1.3.1 Forme et position	12
1.4 Interpolation de courbes	13
1.4.1 Homotopie dans $\mathcal{C}^p(I, X)$	14
1.4.2 Homotopie régulière dans $\mathcal{C}^p(I, X)$	17
1.5 Conclusion	22
2 Un algorithme d'interpolation de courbes polygonales	25
2.1 Du modèle continu au modèle discret	26
2.1.1 La notion de modèle en modélisation géométrique	26
2.1.2 Modèle et classes d'équivalence	27
2.1.3 Un modèle de courbe	30
2.2 Algorithme d'interpolation	32
2.2.1 Schéma principal	32
2.2.2 Construction d'un repère adapté à une courbe échantillonnée	33
2.2.3 Interpolation des repères (première version)	34
2.2.4 Interpolation des repères (seconde version)	37
2.3 Extension	39
2.3.1 Torsion des repères	39
2.3.2 Contrôle de la vitesse de transformation	40
2.3.3 Redressement automatique	40

2.4	Implémentation et résultats	41
2.5	Conclusion	44
II	Aide à la création d'axe	47
1	Squelettisation et axe médian	51
1.1	Axe médian et morphologie mathématique	52
1.1.1	Algorithmes de construction de l'axe médian	53
1.2	Conclusion	54
2	Vers un diagramme de Reeb	57
3	Un algorithme de découpage en bandelettes	63
3.1	Introduction	63
3.2	Géodésiques sur les polyèdres	64
3.3	Fonctions numériques sur les polyèdres	66
3.3.1	Coloriage et caractéristique d'Euler-Poincaré	68
3.3.2	Courbes de niveau des coloriages	71
3.3.3	Courbes de niveau et coloriage sur les triangulations	73
3.3.4	Décomposition en bandelettes	77
3.3.5	Cas des polyèdres non triangulés	78
3.3.6	Cas des triangulations non coloriées	79
3.4	Application au coloriage par la distance géodésique	79
3.4.1	Algorithme	82
3.4.2	Le cas pratique	87
3.5	Création d'axe	89
3.5.1	Quelques résultats	90
3.5.2	À propos du centrage	92
3.6	Conclusion	96
III	Métamorphoses	99
1	État de l'art	103
1.1	L'approche volumique	103
1.1.1	Métamorphoses bidimensionnelles	103
1.1.2	Métamorphoses tridimensionnelles	105
1.2	L'approche surfacique	108

1.2.1	Métamorphoses de contours	109
1.2.2	Métamorphoses de surfaces	112
1.2.3	Métamorphoses et évolutions de surfaces	114
1.3	Conclusion	115
2	Un algorithme de métamorphose	117
2.1	Schéma principal	117
2.2	Système de coordonnées cylindriques	119
2.2.1	Système de coordonnées associé à un modèle de courbe	121
2.2.2	Attachement d'un objet à un modèle de courbe	122
2.2.3	Extensions aux extrémités d'une courbe	124
2.3	Paramétrage et échantillonnage	126
2.3.1	Paramétrage à partir d'un axe	126
2.3.2	Échantillonnage à partir d'un axe	128
2.4	Recalage	132
2.5	Mise en correspondance et interpolation	137
2.5.1	Mise en correspondance	137
2.5.2	Interpolation	138
2.6	Exemples de métamorphoses	141
2.7	Conclusion	146
	Conclusion	147
	Appendices	151
	A Linéarisation d'un champ invariant	151
	B Γ_{ε}^n converge uniformément vers Γ	153
	C Solution numérique de $R' = R \Omega$	155
	D Interpolation des repères (seconde version)	159
	E La topologie en tout ou rien	163
	F Construction d'un champ sur un triangle	167
F.1	Construction d'un champ particulier	167
F.2	Renormalisation	170
	Lexique	173

Bibliographie

179

Introduction

Motivation

Nous abordons ici le calcul automatique de métamorphoses entre objets surfaciques. Pour nous, une métamorphose désigne le passage continu d'une forme à une autre. En particulier, il faut distinguer ce sens de celui, également commun, d'évolutions biologique. Or, si la simulation informatique d'évolution trouve des applications dans l'étude des sciences botaniques, par exemple, celle de métamorphose a par contre bien du mal à se justifier "scientifiquement". Car les métamorphoses, comme nous l'entendons, n'existent que dans notre imaginaire, et c'est bien là leur intérêt. Les cinéastes l'ont bien vite perçu qui dès l'apparition du cinématographe ont tentés de donner forme à cet imaginaire. Les moyens techniques aidant, l'informatique en tête, l'emploi des métamorphoses à l'écran s'est multiplié, offrant au spectateur des images aux contours de plus en plus précis, au risque d'y perdre un peu de poésie.

N'est-il pas étrange de commencer par dénoncer un sujet apparemment non-scientifique et dont les implications pourraient au mieux réduire encore l'espace imaginaire bien étroit laissé par notre société moderne et technologique? À la vérité, nous espérons ces critiques infondées et ceci pour deux raisons. D'abord parce que nous croyons que ce qui permet de qualifier un texte de scientifique ce ne sont pas tant ces implications ou le fait qu'il s'inscrive dans telle grande théorie, que son caractère rigoureux et avant tout honnête et novateur. Ensuite, parce que nous n'avons nullement la prétention de rivaliser avec la bonne vieille "technique" du flou artistique probablement bien plus efficace et chargée de mystères que les transformations stériles d'un Michael Jackson.

Notre travail s'inscrit dans une discipline nouvelle, née de l'informatique, l'*image de synthèse*. Comme son nom l'indique, cette discipline s'intéresse à la production d'images "synthétiques" qui ne pourraient être réalisées sans le support de l'ordinateur. Notre objectif est de définir puis de proposer des outils informatiques qui permettent à des personnes douées d'un certain sens esthétique ou artistique de créer des images fixes ou animées. L'intérêt même de ces images, c'est-à-dire leur potentiel d'expression, reste en grande partie à démontrer,

mais là n'est pas notre rôle. Notre motivation est venue d'une part de l'intuition que des rapprochements entre diverses disciplines telles que l'informatique, l'algorithmique ou les mathématiques et les arts graphiques (au sens large) pouvaient être la source de problèmes riches et passionnants, d'autre part d'un sentiment après tout bien naturel : la curiosité.

Principes et Organisation

Nous nous sommes fixé pour tâche de concevoir un outil informatique permettant d'effectuer des métamorphoses d'objets surfaciques (nous ne nous intéresserons ici qu'au cas des polyèdres ou des surfaces implicites). Pour qu'un outil informatique mérite cette appellation, il doit comporter un certain degré d'automatisation évitant à l'utilisateur potentiel des spécifications souvent fastidieuses voire irréalisables par leur nombre. Paradoxalement, de par ce caractère automatique, un outil définit implicitement une réalisation "canonique" de la tâche à effectuer et prive l'utilisateur d'une quelconque liberté d'expression. Nous pouvons cependant assouplir cette automatisation à l'aide de paramètres intuitifs qui permettent de réaliser une même tâche de manières différentes selon des critères déterminés par l'utilisateur. De ce point de vue, la conception d'un outil de métamorphose est particulièrement ardue car il n'existe pas de moyen a priori pour paramétrer des transformations entre objets.

Comme souvent dans ce type de situations nous pouvons procéder par simplification en nous plaçant dans un cadre beaucoup plus restreint, empruntant aux mathématiques leurs capacités d'abstraction et de simplification.

Nous supposerons ainsi que nos objets informatiques représentent des variétés compactes de dimension 2, plongées ou immergées dans \mathbb{R}^3 . Une métamorphose peut alors s'identifier à une homotopie de plongement ou d'immersion. Ceci suppose des topologies identiques pour deux objets à métamorphoser. Comme les topologies de nos objets sont dénombrables (toute variété compacte de dimension 2 est homéomorphe soit à un tore à n trous soit au recollement de n espaces projectifs) nous pouvons imaginer, dans un premier temps, que les changements topologiques surviennent à des instants isolés où l'on effectue certaines opérations topologiques. En dehors de ces instants nous devons effectivement réaliser des homotopies.

Plus précisément, nous nous sommes intéressés à des objets dont la forme tubulaire permet de les identifier "naturellement" à des cylindres fermés par deux petits disques.

Le cylindre est en effet un objet simple qui s'identifie, sur le plan topologique, au produit $S^1 \times D^1$ du cercle S^1 par le disque D^1 (i.e. un segment); nous pouvons "dé-symétriser" ce produit en considérant le cylindre comme un espace fibré de base D^1 dont les fibres sont isomorphes à S^1 . D^1 peut ainsi être considéré comme l'axe du cylindre auquel sont attachées ses fibres. L'identification naturelle de nos objets signifie que cet axe et ces fibres peuvent

être obtenus naturellement par une construction géométrique.

En paramétrant les axes associés à deux objets cylindriques, nous pouvons mettre en correspondance les fibres des deux objets qui se projettent sur leur axe respectif en un point de même paramètre. Une métamorphose entre deux objets peut alors être conçue comme une transformation entre les axes de ce cylindre suivie d'une transformation entre les paires de fibres associées.

Dans les trois parties qui composent ce mémoire nous verrons ainsi comment transformer un axe en un autre, comment construire un axe à partir d'une forme tubulaire et, finalement, comment réaliser une métamorphose entre objets tubulaires. Plus précisément :

- Dans la première partie, pour définir une transformation entre deux axes, nous construisons une homotopie régulière entre deux segments immergés et nous en déduisons un algorithme d'interpolation entre deux courbes polygonales. Dans cette partie nous insistons particulièrement sur la suite de raisonnements qui mènent de l'intuition mathématique à l'implémentation de l'algorithme.
- Dans la seconde partie nous étudions comment associer une courbe squelette à un objet polyédrique. Dans ce cadre nous proposons une décomposition des polyèdres à l'aide de cycles d'arêtes comparables aux fibres évoquées plus haut.
- Dans la troisième partie nous décrivons un algorithme de métamorphose entre deux objets de forme tubulaire. Des problèmes liés à la définition d'un système de coordonnées cylindriques y sont notamment détaillés dans une optique algorithmique.

Dans la première partie, tout particulièrement, nous emprunterons à la géométrie différentielle un peu de son vocabulaire. Comme ce vocabulaire n'est pas forcément celui de la communauté de l'image de synthèse, nous avons jugé utile d'ajouter en fin de ce mémoire un lexique reprenant les principaux termes employés.

Première partie

Interpolation de courbes

Dans cette première partie, nous nous intéressons au problème de l'interpolation dans \mathbb{R}^3 de deux courbes gauches non-fermées, modélisées par des courbes polygonales.

Les déformations et transformations de courbes ont largement été étudiées en analyse d'image. Ainsi, la recherche des lignes de contours dans une image composée de pixels a conduit à développer un modèle de *contours actifs* [KWT87]. Dans ce cadre le problème est de trouver une courbe suffisamment régulière vérifiant au mieux certaines contraintes liées aux résultats recherchés dans l'image. Comme la définition de cette courbe n'est en général pas explicite, la solution proposée est de faire évoluer une courbe initialement donnée et supposée proche de la solution. Cette évolution, guidée par les contraintes, permet de trouver la courbe optimale. La résolution à l'aide de l'ordinateur fait appel à des techniques d'analyse fonctionnelle et de calcul numérique. Dans ce cadre elle est particulièrement adaptée à des calculs dans des espaces discrets et tout particulièrement dans le plan \mathbb{Z}^2 .

Dans un cadre plus général on étudie dans [KTZ92, ST94, Fau93] l'évolution de courbes fermées en fonction d'une équation d'évolution faisant intervenir des quantités propres à un *plongement* de la courbe dans le plan (euclidien, affine ou projectif), c'est à dire essentiellement la courbure de la courbe et ses dérivées. Dans [Fau93], on introduit la méthode du repère mobile pour analyser les déformations d'une courbe. Nous reprendrons par la suite cette idée.

Ces techniques ne sont cependant pas applicables à notre problématique car :

- les objets étudiés sont différents : il s'agit de courbes de \mathbb{R}^3 et non du plan,
- la problématique est différente : il s'agit pour nous de faire évoluer une courbe entre des positions initiale et finale *données*,
- les objets ne sont pas représentés par les points d'une grille (le plan \mathbb{Z}^2) mais par des n-uplets de réels (c.a.d des courbes polygonales).

Quelques approches ont été proposées en infographie pour transformer une ligne polygonale en une autre [SG92, SGWM93, GH94, SR95, GG95] afin de réaliser une métamorphose ou de calculer des images intermédiaires entre deux images clés d'une animation (key-framing). Dans ces méthodes les polygones sont supposés planaires et fermés, ce qui n'est pas notre cas. Nous aurons l'occasion cependant de revenir sur les deux dernières références.

Il nous faut ainsi définir entièrement un procédé d'interpolation de courbes de \mathbb{R}^3 . Dans un premier temps nous considérerons ce problème dans le cadre classique de la géométrie différentielle et nous proposerons une solution compatible avec nos besoins. Nous verrons

ensuite comment adapter cette solution dans un cadre algorithmique où les objets étudiés ne sont plus des variétés différentielles mais des objets décrits par une combinatoire.

L'idée directrice sera de définir des procédés les plus simples possibles ne faisant appel, si possible, qu'à des notions intrinsèques aux objets étudiés. Dans les sections suivantes nous progresserons vers une solution en écartant au fur et à mesure certains procédés intuitifs d'interpolation et en montrant pourquoi ils ne peuvent fonctionner.

Chapitre 1

Un procédé canonique d'interpolation de courbes sur les groupes de Lie

1.1 Interpolation de points

Avant de se poser le problème de l'interpolation de courbes nous pouvons déjà regarder comment interpoler deux points P_0 et P_1 : nous cherchons à relier P_0 à P_1 par un chemin. Pour donner un sens à cette quête, nous supposons que ces points appartiennent à un espace topologique donné X .

- Si nous n'avons aucune information supplémentaire sur l'espace X , ce problème peut ne pas avoir de solution. C'est le cas par exemple si P_0 et P_1 sont dans deux composantes connexes distinctes de X .
- Si nous supposons X connexe par arc alors nous savons que P_0 et P_1 peuvent être reliés mais nous n'avons aucun moyen de définir un chemin particulier.
- Si X est de plus muni d'une métrique nous pouvons tenter de chercher notre solution parmi les chemins γ les plus courts. Rappelons que la longueur d'un chemin [LFA77] est donnée par

$$\sup_{\sigma \in \mathcal{S}} \sum_{0 \leq i < n} d(\gamma(t_i), \gamma(t_{i+1}))$$

où \mathcal{S} est l'ensemble des subdivisions $\sigma = (t_0, t_1, \dots, t_n)$ de $[0,1]$ ($t_0 = 0, t_n = 1$) et d la distance sur X . Lorsque cette quantité est finie on dit que γ est *rectifiable*.

Malheureusement, dans le cas général, nous ne savons pas s'il existe un chemin rectifiable joignant deux points et quand il en existe un, s'il existe un unique chemin rectifiable de longueur minimale.

Aux vues des problèmes soulevés par ces quelques exemples nous proposons une définition du mot interpolation.

Définition 1.1 *Un procédé d'interpolation sur un espace topologique X est un moyen (canonique) pour construire une application γ continue de $[0,1]$ dans X , d'extrémités $\gamma(0) = P_0$ et $\gamma(1) = P_1$, où P_0 et P_1 sont deux points quelconques de X .*

L'exemple fondamental est, à ce titre, celui de l'interpolation linéaire sur les espaces vectoriels (topologiques).

Si l'on se restreint à une interpolation *locale*, les points P_0 et P_1 étant supposés proches, il est également possible de définir des procédés d'interpolation sur deux autres catégories d'espaces topologiques : les groupes de Lie et les variétés riemanniennes.

Dans ces deux cas, on peut en effet définir une paramétrisation locale canonique (l'application exponentielle). L'inverse de cette paramétrisation envoie localement la variété sur des ouverts (convexes) d'un espace vectoriel où l'on peut utiliser l'interpolation linéaire.

Sur les variétés riemanniennes ce procédé consiste à relier P_0 et P_1 par une géodésique de longueur minimale. Nous savons en effet qu'autour de tout point il existe un voisinage dans lequel tout couple de points est relié par une unique géodésique de longueur minimale. Ce voisinage est appelé voisinage totalement normal dans [Car93]. La proximité des points est nécessaire comme le montre l'exemple de deux points diamétralement opposés sur une sphère munie de sa métrique usuelle, puisqu'il existe une infinité de géodésiques de longueur minimale (des demi-arcs de grand cercles) reliant ces points. Notons tout de même que si X est connexe et complet (comme espace topologique) le théorème de Hopf-Rinow [Car93] nous assure l'existence d'au moins un plus court chemin pour tout couple de points.

Ces considérations peuvent paraître éloignées de notre but, interpoler des courbes pour l'infographie, mais nous verrons par la suite qu'elles permettent effectivement dans des cas bien précis de définir une interpolation de courbe canonique simplifiant au maximum l'interaction avec l'utilisateur.

Après avoir défini un mode d'interpolation pour les points nous pouvons voir si cette interpolation s'étend simplement aux courbes. Rappelons tout d'abord ce que nous entendons par le mot courbe.

1.2 Définition d'une courbe

Le mot "courbe" désigne pour nous un objet bien précis qui est distinct du sens usuel. En effet, dans les livres de géométrie différentielle classique [BG87] une courbe est une variété différentielle de dimension 1, connexe, de classe \mathcal{C}^1 au moins. On sait d'ailleurs ([BG87], p

127) qu'une telle courbe est difféomorphe soit à la droite réelle soit au cercle et que dans ce cas elle est compacte.

Mais nous utiliserons le mot courbe dans le sens d'arc paramétré :

Un arc paramétré de classe \mathcal{C}^p ($p \geq 1$) d'une variété X est la donnée d'une immersion $\gamma \in \mathcal{C}^p(]0, 1[, X)$.

La condition d'immersion assure qu'on peut définir l'espace tangent à la courbe et sa vitesse pour tout *paramètre* $t \in]0, 1[$ mais n'interdit pas les autointersections.

Pour être plus précis nous utiliserons une définition restreinte plus adaptée à notre problème, à savoir un *arc compact* [LFA77], l'arc paramétré étant défini sur le segment $[0, 1]$ et non $]0, 1[$ (une courbe est donc l'immersion d'une variété de dimension 1 à bord, le bord étant constitué de ses deux extrémités, c'est à dire de la sphère S^0).

Remarquons que si X est un espace vectoriel, les courbes *ne forment pas* un sous-espace vectoriel de l'espace $\mathcal{C}^p([0, 1], X)$. Ainsi la somme de deux courbes n'est a priori pas une courbe.

Ayant défini les objets de travail nous pouvons poser le problème de l'interpolation de courbes.

1.3 Position du problème

Le problème (P_c) de l'interpolation de courbes se pose ainsi :

Soient γ_0 et γ_1 deux courbes de classe \mathcal{C}^p , comment définir une interpolation $\gamma_t, t \in [0, 1]$ telle que :

- $\gamma_{t=0} = \gamma_0, \gamma_{t=1} = \gamma_1,$
- *pour tout t, γ_t est une courbe de classe $\mathcal{C}^p,$*
- *l'application $(t, s) \mapsto \gamma_t(s)$ est continue.*

En particulier une interpolation suppose les deux courbes γ_0 et γ_1 homotopes (si nous voulions interpoler des courbes fermées sur un tore, cette dernière remarque a le mérite de montrer les limites de cette interpolation...).

Comme les courbes sont données par des paramétrisations, nous avons une correspondance naturelle $\gamma_0(s) \mapsto \gamma_1(s)$. On pourrait donc penser à définir $\gamma_t(s)$ par l'interpolation des points $\gamma_0(s)$ et $\gamma_1(s)$ comme proposé dans la section 1.1. Les exemples de la figure 1.1

montrent cependant que cette construction n'est pas acceptable : dans la figure de gauche la paramétrisation n'est pas injective, tandis que dans celle de droite elle n'est pas régulière. Ainsi l'extension du problème de l'interpolation des points à celui de l'interpolation

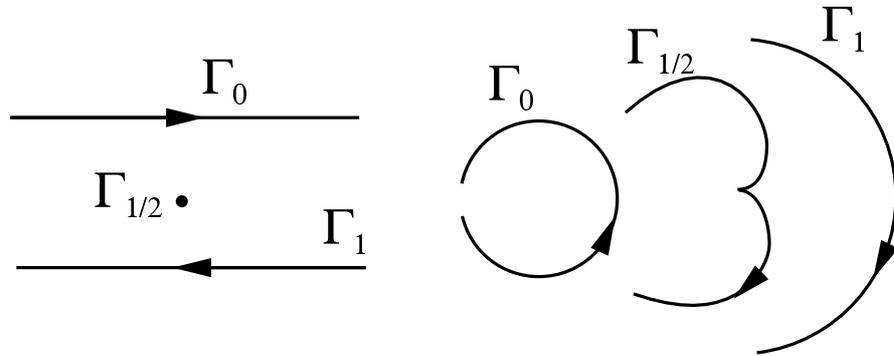


FIG. 1.1 – À gauche : interpolation au temps $1/2$ entre deux segments Γ_0 et Γ_1 du plan orientés dans le sens contraire. À droite : interpolation au temps $1/2$ entre deux arcs de cercle. La courbe interpolée est un arc de cardioïde.

des courbes n'est pas si immédiate qu'elle y paraît.

Avant de proposer une solution plus satisfaisante nous présentons rapidement dans la section 1.3.1 une autre approche. Nous supposons temporairement que l'espace ambiant X est l'espace euclidien \mathbb{R}^3 . Le fait de nous restreindre à \mathbb{R}^3 nous permet de définir la forme d'une courbe par l'intermédiaire d'un groupe agissant sur l'espace. Ce groupe (géométrique) est ici le groupe des déplacements.

Nous caractériserons donc une courbe par sa forme et sa position.

1.3.1 Forme et position

Si on fait agir le groupe des déplacements de \mathbb{R}^3 sur une courbe γ on obtient un ensemble de courbes (l'orbite de γ par l'action des déplacements).

Cette classe de courbes constitue la **forme** de γ .

Une **position** est un élément particulier de cette classe ou encore un déplacement.

Pour interpoler une courbe nous pouvons ainsi tenter d'interpoler séparément sa forme et sa position. Or la forme d'une courbe de \mathbb{R}^3 est déterminée par deux fonctions appelées courbure et torsion. Plus précisément nous avons le théorème de représentation suivant ([BG87], p 339) :

Théorème 1.1 Soit I un intervalle de \mathbb{R} , $\kappa \in C^1(I; \mathbb{R}_+^*)$ et $\tau \in C^0(I; \mathbb{R})$, et $\mathbf{u}, \mathbf{v}, \mathbf{w}$, trois

points de E espace vectoriel euclidien de dimension 3. Il existe alors un arc géométrique C orienté de classe C^3 (nécessairement birégulier¹) de E , et un seul dont une paramétrisation normale (I, γ) soit telle que

$$\gamma(0) = \mathbf{u}, \gamma'(0) = \mathbf{v}, \frac{\gamma''(0)}{\|\gamma''(0)\|} = \mathbf{w}$$

et que κ et τ soient respectivement la courbure et la torsion de C en $\gamma(t)$ pour tout $t \in I$.

Ce théorème se généralise d'ailleurs en dimension quelconque ([Die71b], p 323).

Cet énoncé s'applique à des courbes paramétrées par l'abscisse curviligne (i.e données par une paramétrisation normale). Or dans notre cas, c'est l'intervalle de paramétrisation qui est fixé et non la vitesse. Il faut donc ajouter la vitesse (qui détermine la paramétrisation) à la courbure et à la torsion pour définir la forme de nos courbes.

Pour interpoler la forme d'une courbe nous pouvons interpoler séparément chacune de ces quantités. Cependant l'ensemble des triplets (v, κ, τ) de fonctions qui représentent une forme ne constitue pas un espace vectoriel. Par exemple, il n'existe pas de courbes à courbure nulle et torsion non nulle. Le théorème de représentation précédent ne s'applique d'ailleurs qu'aux courbes birégulières. Les triplets de fonctions correspondantes constituent certes un ensemble convexe, mais il serait dommage de ne pouvoir intégrer les segments de droite dans nos interpolations...

La notion de forme précédemment décrite n'est finalement pas très bien adaptée à notre problème. Intuitivement cela peut s'interpréter par le fait que les déplacements n'opèrent pas simplement (transitivement) sur chacune des orbites de courbes comme le montre le cas du segment de droite, puisque toutes les rotations admettant un axe colinéaire au segment et un centre sur la droite support de ce segment le laisse invariant.

Dans la section suivante nous développons un procédé d'interpolation de courbes qui remédie à ce problème. Pour cela nous nous plaçons dans un cadre plus général où nos courbes sont des arcs compacts à valeurs dans un groupe de Lie. Ceci nous permet de représenter les courbes comme des lignes intégrales de certains champs non singuliers. L'interpolation de ces champs fournit alors l'interpolation des courbes.

1.4 Interpolation de courbes

Dans toute cette partie I désigne l'intervalle compact $[0, 1]$, X une variété connexe de dimension finie n et de classe C^p , $p > 2$ et $C^p(I, X)$ l'ensemble des applications de I dans X de classe C^p .

1. un arc est dit *birégulier* s'il admet une paramétrisation γ telle que γ' et γ'' sont indépendants

Cette section est structurée de la façon suivante :

- En premier lieu, dans la section 1.4.1, nous décrivons comment interpoler deux fonctions γ_0 et γ_1 de $\mathcal{C}^p(I, X)$,
- puis, dans la section 1.4.2, nous montrerons comment étendre ce procédé d'interpolation à des courbes, c'est-à-dire à des *immersions* du segment I (i.e. $\gamma'_0 \neq 0$ et $\gamma'_1 \neq 0$). Cette extension requiert la donnée d'un repère mobile adapté à chaque immersion. Nous décrivons donc comment choisir un tel repère et nous commenterons l'influence de ce choix lorsque X est un espace vectoriel euclidien.

1.4.1 Homotopie dans $\mathcal{C}^p(I, X)$

Nous cherchons à définir une interpolation entre deux fonctions, γ_0 et γ_1 , de $\mathcal{C}^p(I, X)$, c'est-à-dire que nous cherchons une famille γ_t de fonctions telles que :

1. $\gamma_t \in \mathcal{C}^p(I, X), \forall t \in I$,
2. $\gamma_{t=0} = \gamma_0$ et $\gamma_{t=1} = \gamma_1$,
3. l'application $(s, t) \mapsto \gamma_t(s)$ est continue de $I \times I$ dans X .

Pour pouvoir se ramener à des interpolations linéaires dans des espaces vectoriels, nous allons considérer γ_0 et γ_1 comme des courbes intégrales de certains champs de vecteurs sur X et interpoler ces champs plutôt que les fonctions elles-mêmes. Pour cela nous associons à chaque fonction γ son **hodographe** c'est-à-dire la fonction de classe \mathcal{C}^{p-1} à valeur dans le fibré tangent TX :

$$s \mapsto \gamma'(s) = T_s\gamma \cdot 1_{\mathbb{R}}.$$

Cet hodographe est donc le *relèvement canonique de γ dans TX* .

Puis nous interpolons les hodographes de γ_0 et γ_1 et nous en déduisons un champ de vecteurs \mathbf{V}_t sur X dépendant du temps.

Par ailleurs, comme X est connexe, nous supposons donné un chemin $t \mapsto \gamma_t(0)$ joignant $\gamma_0(0)$ à $\gamma_1(0)$.

γ_t est alors obtenu en intégrant \mathbf{V}_t sur I avec la condition initiale $\gamma_t(0)$.

Ce principe d'interpolation soulève plusieurs questions :

1. Comment comparer des vecteurs de TX qui sont dans des espaces tangents (i.e des fibres) différents ?
2. Comment interpoler des hodographes ?

3. Comment construire à partir d'un hodographe un champ de vecteurs dépendant du temps sur X ?
4. Comment être sûr que ce champ est intégrable ?

En répondant point par point à ces questions, nous verrons qu'il est nécessaire de faire des hypothèses supplémentaires sur X et que la notion de groupe de Lie permet de s'assurer de ces hypothèses.

1. Comparaison entre vecteurs tangents : afin de comparer des vecteurs qui n'appartiennent pas à un même espace tangent, ce qui revient à construire un isomorphisme entre espaces tangents, on peut munir la variété X d'un opérateur appelé *connexion*. Généralement cet opérateur ne permet de définir cet isomorphisme que de manière locale (on peut donc obtenir canoniquement une *trivialisatation locale* en tout point de TX). Pour être capable de comparer *globalement* tous les vecteurs de TX nous devons supposer que TX est globalement trivialisable, c'est à dire qu'il existe un difféomorphisme qui envoie TX sur le produit $X \times \mathbb{R}^n$ et qui respecte la projection canonique de TX sur X .

En supposant que X est un groupe de Lie (connexe de dimension n) nous savons que cette trivialisatation est possible et qu'elle peut être définie canoniquement.

Dans ce cas, en effet, chaque élément x du groupe X induit un automorphisme de X , $y \mapsto x^{-1}y$, que nous notons τ_x . Par différentiation, cet automorphisme induit lui-même un isomorphisme entre l'espace T_xX tangent en x et l'espace T_eX tangent en l'unité e du groupe². Nous notons θ_x cet isomorphisme :

$$\theta_x = T_x\tau_x.$$

L'hodographe γ' d'une fonction γ peut ainsi être transporté par θ_γ en T_eX (voir figure 1.2) :

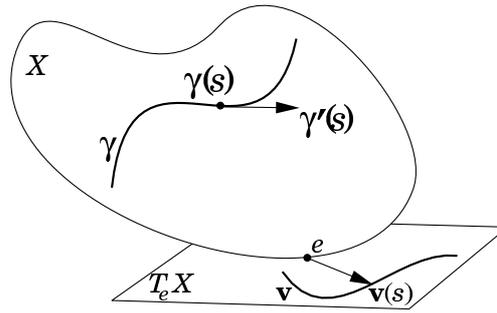
$$\gamma' \mapsto \mathbf{v} = \theta_\gamma \cdot \gamma'. \quad (1.1)$$

2. Interpolation d'hodographes : en nous ramenant à un problème dans T_eX , nous pouvons définir l'interpolation des hodographes par l'interpolation des hodographes transportés.

Les deux fonctions γ_0 et γ_1 fournissent en effet deux hodographes transportés \mathbf{v}_0 et \mathbf{v}_1 en utilisant le transport défini par l'équation 1.1. \mathbf{v}_0 et \mathbf{v}_1 étant à valeurs dans l'espace vectoriel T_eX , nous pouvons alors les interpoler canoniquement à l'aide de l'interpolation linéaire :

$$\mathbf{v}_t(s) = (1 - t) \mathbf{v}_0(s) + t \mathbf{v}_1(s).$$

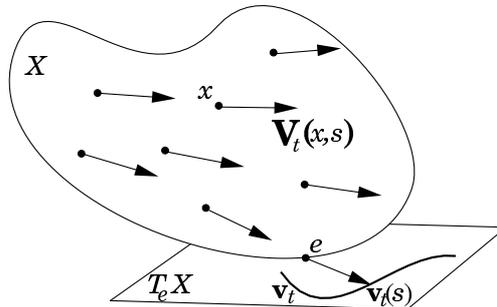
2. On note souvent \mathcal{G}_e cet espace par référence à la structure d'algèbre dont il peut être muni. Comme nous n'utilisons pas cette structure nous préférons conserver la notation plus classique T_eX .

FIG. 1.2 – Transport du vecteur $\gamma'(s)$ en $T_e X$.

3. Construction d'un champ de vecteurs à partir d'un hodographe : un vecteur \mathbf{v} de $T_e X$ fournit un champ invariant pour le groupe, de la forme :

$$\mathbf{V}(x) = \theta_x^{-1} \cdot \mathbf{v}.$$

À partir de la fonction interpolée $s \mapsto \mathbf{v}_t(s)$ on obtient donc par “translation” en tout point un champ de vecteurs dépendant du paramètre³ s (voir figure 1.3) :

FIG. 1.3 – Le vecteur $\mathbf{v}_t(s)$ est transporté en tout point de X .

$$\mathbf{V}_t(x, s) = \theta_x^{-1} \cdot \mathbf{v}_t(s).$$

4. Intégration de \mathbf{V}_t : pour une condition initiale $\gamma_t(0)$ donnée nous devons intégrer le système (de Cauchy) (E_t) suivant :

$$\begin{cases} \gamma'(s) = \mathbf{V}_t(\gamma(s), s) \\ \gamma(0) = \gamma_t(0) \end{cases}$$

³. Notons que la notion de champ dépendant d'un paramètre peut être ignorée si on considère le champ $(\mathbf{V}_t(x, s), 1_I)$ sur la variété produit $X \times I$.

En général, les courbes intégrales d'un système défini par un champ de vecteurs et une condition initiale ne sont pas définies sur tout intervalle :

prenons sur \mathbb{R} l'exemple classique du champ $\mathbf{V}(x) = -x^2$. Le système correspondant de condition initiale x_0 a pour solution $s \mapsto \frac{x_0}{1+sx_0}$ qui est définie sur $]\frac{-1}{x_0}, +\infty[$ si x_0 est positif et $]-\infty, \frac{-1}{x_0}[$ sinon.

Afin de s'assurer que le système (E_t) a une solution, nous pouvons supposer que X est compact, ou bien que X admet une *représentation linéaire fidèle*, c'est à dire qu'il existe un morphisme injectif de groupe de Lie envoyant X dans $GL(n, \mathbb{R})$, le groupe linéaire des matrices inversibles de taille $n \times n$.

En effet,

- si X est compact, le problème de Cauchy (E_t) est intégrable sur I (c.f. [Mal72], §I.2.2),
- si X admet une représentation linéaire fidèle, alors nous montrons en annexe A que le système (E_t) se "transforme" en un système linéaire et la théorie des équations différentielles linéaires nous dit que ce système est intégrable (voir par exemple [BG87], §1.6).

Nous avons ainsi défini sur tout groupe de Lie connexe de dimension finie n qui est soit compact, soit représentable fidèlement dans $GL(n, \mathbb{R})$, un procédé canonique d'interpolation de deux fonctions γ_0 et γ_1 de $\mathcal{C}^p(I, X)$. Ce procédé nécessite la donnée d'un chemin reliant $\gamma_0(0)$ à $\gamma_1(0)$. Lorsque ces points sont suffisamment proches, on peut les relier, comme proposé dans la section 1.1, à l'aide de l'application exponentielle en $\gamma_0(0)$ qui est un difféomorphisme local (c.f. [Die71b], §19.8.6) de $T_{\gamma_0(0)}X$ sur X .

1.4.2 Homotopie régulière dans $\mathcal{C}^p(I, X)$

Nous cherchons désormais à définir une interpolation entre deux immersions de I de classe \mathcal{C}^p , c'est-à-dire que nous cherchons à résoudre le problème (P_c) de la section 1.3 que nous rappelons ci-dessous :

Comment trouver une famille γ_t de fonctions telles que :

1. γ_t est une immersion de I de classe \mathcal{C}^p pour tout t de I ,
2. $\gamma_{t=0} = \gamma_0$ et $\gamma_{t=1} = \gamma_1$,
3. l'application $(s, t) \mapsto \gamma_t(s)$ est continue de $I \times I$ dans X .

Le principe d'interpolation est une extension du cas précédent. Cependant dans le cas d'immersions, les hodographes transportés \mathbf{v}_0 et \mathbf{v}_1 précédemment définis, ne "passent" pas par

le vecteur nul 0_e de T_eX et \mathbf{v}_t ne doit pas non plus s'annuler. Nous devons donc effectuer une interpolation d'hodographes dans T_eX privé de 0_e . Comme cet ensemble n'est pas convexe nous ne pouvons plus utiliser l'interpolation linéaire pour \mathbf{v}_t . Pour y remédier nous allons remplacer les hodographes transportés par des fonctions à valeurs dans un groupe de Lie, et non plus dans T_eX privé de 0_e . Nous pourrons ainsi appliquer la méthode d'interpolation précédente à ces fonctions, puis nous en déduisons un hodographe \mathbf{v}_t . À partir de \mathbf{v}_t nous pourrons procéder comme dans la section 1.4.1.

Afin de remplacer les hodographes par d'autres fonctions, nous enrichissons la structure de notre groupe de Lie en le munissant d'une métrique invariante, ce qui revient à munir T_eX d'un produit scalaire. $T_eX \setminus \{0_e\}$ s'identifie alors, comme sous-variété de T_eX , au produit $\mathbb{R}_+^* \times S_e^{n-1}$ par l'application :

$$\mathbf{v} \mapsto (\|\mathbf{v}\|, \frac{\mathbf{v}}{\|\mathbf{v}\|})$$

Nous pourrions alors effectuer l'interpolation séparément sur \mathbb{R}_+^* , qui est convexe, et sur la sphère unité S_e^{n-1} de T_e . Mais, sauf pour des valeurs particulières de n (2 et 4), cette sphère ne peut être un groupe de Lie et nous ne pouvons appliquer la méthode précédente.

Pour cela, nous remplaçons le couple $(\|\mathbf{v}\|, \frac{\mathbf{v}}{\|\mathbf{v}\|})$ associé à une courbe γ (c.f équation 1.1), par un couple $(\|\mathbf{v}\|, r)$, où r correspond au transport dans T_eX d'un repère mobile R adapté⁴ à γ et orthonormé. En effet, après avoir défini un repère adapté R pour une courbe γ nous pouvons transporter ce repère dans T_eX comme pour l'équation de transport 1.1 :

$$R(s) = (\mathbf{e}_1(s), \mathbf{e}_2(s), \mathbf{e}_3(s)) \mapsto r(s) = \theta_{\gamma(s)}.R(s) = (\theta_{\gamma(s)}.\mathbf{e}_1(s), \theta_{\gamma(s)}.\mathbf{e}_2(s), \theta_{\gamma(s)}.\mathbf{e}_3(s)).$$

Le choix d'une métrique invariante sur X nous assure que $\theta_{\gamma(s)}$ est une isométrie et donc que $r(s)$ est orthonormé.

En fixant une base orthonormée de T_eX , nous pouvons identifier $r(s)$ à un élément de $SO(n, \mathbb{R})$. Nous pouvons donc interpoler les repères r_0 et r_1 , correspondant aux courbes γ_0 et γ_1 , sur le groupe de Lie connexe $SO(n, \mathbb{R})$ à l'aide de la première méthode d'interpolation. Remarquons qu'un changement de base dans T_eX se traduit par une multiplication par une matrice orthogonale à coefficients constants et ne modifie pas le résultat de l'interpolation, ce qui montre le caractère intrinsèque de l'interpolation des repères. Par contre, comme précisé dans la première méthode d'interpolation, il est nécessaire de spécifier une courbe $r_t(0)$ reliant $r_0(0)$ à $r_1(0)$.

On en déduit ensuite un hodographe interpolé (dans T_eX) en multipliant le premier vecteur du repère interpolé r_t par la vitesse interpolée v_t . Notons que **l'interpolation linéaire de la vitesse conduit à l'interpolation linéaire de la longueur.**

4. c'est à dire que le premier vecteur de R est $\frac{\gamma'}{\|\gamma'\|}$.

Pour compléter le procédé d'interpolation, il ne nous reste donc plus qu'à décrire comment associer un repère mobile adapté à une courbe tracée sur X .

Détermination d'un repère mobile adapté

Pour caractériser un repère adapté à une courbe, nous allons examiner sa matrice de Cartan.

Si $R = (\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$ est un repère mobile orthonormé et adapté à la courbe γ (i.e. $\mathbf{e}_1 = \frac{\gamma'}{\|\gamma'\|}$), on peut lui associer sa matrice de Cartan $(\omega_{i,j})_{1 \leq i,j \leq n}$ où :

$$\omega_{i,j} = \langle \nabla_{\mathbf{e}_1} \mathbf{e}_i, \mathbf{e}_j \rangle$$

et où ∇ désigne la dérivation covariante associée à la métrique $\langle \cdot, \cdot \rangle$ sur X ([Mal72], §8.4). En particulier, cela signifie que pour tout champ de vecteur $\mathbf{A}, \mathbf{B}, \mathbf{C}$, on a :

$$\mathbf{A} \langle \mathbf{B}, \mathbf{C} \rangle = \langle \nabla_{\mathbf{A}} \mathbf{B}, \mathbf{C} \rangle + \langle \mathbf{B}, \nabla_{\mathbf{A}} \mathbf{C} \rangle \quad (1.2)$$

où $\mathbf{A}f$ désigne la dérivation de la fonction réelle f relativement au champ \mathbf{A} . Comme R est orthonormé, on en déduit que sa matrice de Cartan est antisymétrique.

Pour choisir un repère adapté nous pouvons ainsi imposer une forme particulière de matrice antisymétrique et voir si cette matrice peut être la matrice de Cartan d'un repère de γ . Nous donnons et commentons deux exemples classiques :

Le repère de Frenet : nous choisissons une matrice antisymétrique de la forme :

$$\begin{aligned} \omega_{i,j} &= 0 & \text{si } |i - j| \neq 1 \\ \omega_{i,i+1} &> 0 \end{aligned}$$

Dans [Die71b], p 323, on montre qu'il existe "en général" un unique repère adapté à γ qui admette une telle matrice de Cartan. Dans ce cas, ce repère est le repère de Frenet de γ et $\omega_{i,i+1}$ est appelé la $i^{\text{ème}}$ courbure de γ . Dans \mathbb{R}^3 , par exemple, "en général" signifie que γ est birégulière.

Le repère du transport relativement parallèle : un autre repère très employé en infographie est le repère du transport relativement parallèle, ou RTRP, introduit dans [Bis75] dans le cas des espaces euclidiens. Par rapport au repère de Frenet il a l'avantage d'être défini pour toute courbe immergée. Par contre il est défini à un paramètre près : la donnée d'un repère adapté "initial" en un point de la courbe (par exemple en une extrémité).

Nous définissons le RTRP par sa matrice de Cartan, puis nous montrons l'existence de tels repères pour toute courbe γ .

Définition 1.2 *Un repère orthonormé et adapté à une courbe γ est un RTRP si sa matrice de Cartan est de la forme :*

$$\omega_{i,j} = 0 \quad \text{si } i \neq 1 \text{ et } j \neq 1.$$

Ceci signifie que, pour tout $i \neq 1$, $\nabla_{\mathbf{e}_1} \mathbf{e}_i$ est parallèle à \mathbf{e}_1 , d'où la dénomination de ce repère. En utilisant l'équation 1.2 de compatibilité de ∇ avec la métrique, on voit que l'on a nécessairement :

$$\nabla_{\mathbf{e}_1} \mathbf{e}_i = -\langle \nabla_{\mathbf{e}_1} \mathbf{e}_1, \mathbf{e}_i \rangle \mathbf{e}_1. \quad (1.3)$$

Remarquons que cette équation est une extension de l'équation

$$f' = -\langle \gamma'', f \rangle \gamma'$$

proposée dans [Klo86]. Elle montre en particulier que le transport relativement parallèle, ou TRP, coïncide avec le transport parallèle (i.e $\nabla_{\mathbf{e}_1} \mathbf{e}_i = 0$) si et seulement si γ est une géodésique.

Remarque 1.1 *Le TRP conserve les angles (entre deux champs relativement parallèles ou bien entre un tel champ et \mathbf{e}_1) et la norme. C'est une conséquence directe de l'équation de compatibilité 1.2 et de l'équation 1.3.*

De plus l'équation 1.3 permet effectivement de définir un repère pour une condition initiale donnée.

Proposition 1.1 *Pour toute courbe γ et tout repère R_0 adapté à $\gamma(0)$, il existe un unique RTRP R tel que $R(0) = R_0$.*

Démonstration : l'existence et l'unicité d'un champ relativement parallèle, pour une condition initiale donnée, se montre de la même manière que pour un champ parallèle (c.f. [Fer88], p 140) :

En tout point de γ , on peut par une carte locale de X , retranscrire l'équation 1.3 en une équation différentielle linéaire du premier ordre de la forme

$$\frac{d\mathbf{E}_i}{ds} = A \cdot \mathbf{E}_i$$

où \mathbf{E}_i représente \mathbf{e}_i dans la carte, et A est une matrice carrée dont les coefficients font intervenir les composantes de $\frac{d\mathbf{E}_1}{ds}$ et ce qu'on appelle classiquement les symboles de Christoffel associés à ∇ dans la carte choisie. L'existence du TRP est donc assurée localement.

Nous pouvons ensuite procéder comme dans [Bis75] : comme l'image d'un arc compact est compacte, nous pouvons la recouvrir par un nombre fini de cartes et raccorder les solutions trouvées dans chaque carte. L'unicité pour une condition initiale donnée découle alors de la propriété de conservation des angles de la remarque 1.1 \square

Remarque 1.2 *Le RTRP est caractérisé par une matrice de Cartan de norme minimale.*

Démonstration : si $R = (\mathbf{e}_1, \dots, \mathbf{e}_n)$ est un repère mobile orthonormé et adapté à la courbe γ , et $(\omega_{i,j})_{1 \leq i,j \leq n}$ est sa matrice de Cartan, nous avons :

$$\sum_{i,j} \omega_{i,j}^2 = \sum_{i=1 \text{ ou } j=1} \omega_{i,j}^2 + \sum_{i \neq 1 \text{ et } j \neq 1} \omega_{i,j}^2$$

Comme $\omega_{1,i} = -\omega_{i,1} = \langle \nabla_{\mathbf{e}_1} \mathbf{e}_1, \mathbf{e}_i \rangle$ et que R est orthonormé, on a

$$\sum_{i,j} \omega_{i,j}^2 = 2 \|\nabla_{\mathbf{e}_1} \mathbf{e}_1\|^2 + \sum_{i \neq 1 \text{ et } j \neq 1} \omega_{i,j}^2$$

La définition 1.2 du RTRP permet de conclure \square

Lien entre l'interpolation et les matrices de Cartan

On peut se demander si cette interpolation revient à interpoler les matrices de Cartan des repères, c'est à dire si

$$r' = r.(\omega_{i,j}).$$

La réponse est négative dans le cas général.

En effet, en supposant pour simplifier que γ est paramétrée par l'abscisse curviligne, les coefficients de $r^{-1}.r'$ sont de la forme :

$$\Omega_{i,j} = \langle (\theta_\gamma \cdot \mathbf{e}_i)', \theta_\gamma \cdot \mathbf{e}_j \rangle$$

tandis que

$$\omega_{i,j} = \langle \nabla_{\mathbf{e}_1} \mathbf{e}_i, \mathbf{e}_j \rangle = \langle \theta_\gamma \cdot \nabla_{\mathbf{e}_1} \mathbf{e}_i, \theta_\gamma \cdot \mathbf{e}_j \rangle$$

Or dans le cas général $\Omega_{i,j} \neq \omega_{i,j}$ même si R est un RTRP. On peut le vérifier par exemple dans $SO(3, \mathbb{R})$ sur des trajectoires non géodésiques.

Le cas euclidien : par contre, lorsque X est un espace euclidien on a toujours $\Omega_{i,j} = \omega_{i,j}$. Ceci entraîne des propriétés intéressantes pour notre interpolation.

Propriété 1.1 *Si les deux courbes γ_0 et γ_1 sont munies du RTRP (resp. du repère de Frenet) alors la courbe interpolée est automatiquement munie du RTRP (resp. du repère de Frenet).*

Propriété 1.2 *De plus, la courbure interpolée est majorée par une fonction monotone joignant la courbure κ_0 de γ_0 à la courbure κ_1 de γ_1 .*

Démonstration :

1. En effet, les matrices de Cartan des repères de Frenet ou bien des RTRPs forment des ensembles convexes dans l'ensemble des matrices de Cartan (et même un espace vectoriel dans le cas des RTRPs), l'interpolation linéaire de ces matrices fournit donc une matrice de même type.
2. De plus la courbure d'une courbe γ vaut par définition :

$$\kappa = \|\nabla_{\mathbf{e}_1} \mathbf{e}_1\|.$$

Si γ_0 et γ_1 sont paramétrées par l'abscisse curviligne, le premier vecteur de la matrice de Cartan est justement le vecteur $\nabla_{\mathbf{e}_1} \mathbf{e}_1$ exprimé dans le repère type correspondant, on en déduit à l'aide de l'inégalité triangulaire que la courbure κ_t de la courbe interpolée vérifie :

$$\kappa_t \leq (1-t)\kappa_0 + t\kappa_1.$$

Cette inégalité devient d'ailleurs une égalité vérifiée par *toutes* les courbures dans le cas du repère de Frenet.

Si γ_0 et γ_1 ne sont pas paramétrées par l'abscisse curviligne nous devons tenir compte de la vitesse. Nous avons en effet :

$$\mathbf{e}'_i(s) = \nabla_{\gamma'(s)} \mathbf{e}_i(s) = \nabla_{\|\gamma'(s)\| \mathbf{e}_1(s)} \mathbf{e}_i(s) = \|\gamma'(s)\| \nabla_{\mathbf{e}_1(s)} \mathbf{e}_i(s)$$

ce qui montre que

$$\Omega_{i,j} = v \omega_{i,j}$$

où l'on a posé $v = \|\gamma'\|$. L'interpolation se traduit donc par une interpolation du produit de la vitesse par la(les) courbure(s). Comme la vitesse est interpolée linéairement on a

$$\kappa_t \leq \frac{(1-t)v_0\kappa_0 + tv_1\kappa_1}{v_t} = \kappa_0 + \frac{1}{\left(1 - \frac{v_0}{v_1}\right) + \frac{1}{t} \frac{v_0}{v_1}} (\kappa_1 - \kappa_0), \quad t \neq 0$$

ce qui montre que l'interpolation de la courbure est majorée par une fonction monotone valant κ_0 en 0 (resp. κ_1 en 1) (en particulier $\kappa_t \leq \max(\kappa_0, \kappa_1)$).

□

1.5 Conclusion

Nous avons ainsi défini sur tout groupe de Lie connexe, compact ou représentable fidèlement dans un groupe linéaire, et muni d'une métrique invariante, un procédé d'interpolation entre deux immersions γ_0 et γ_1 du segment $[0, 1]$. Nous avons pour cela muni chaque courbe γ_0

et γ_1 d'un repère mobile adapté R_0 et R_1 . En considérant les matrices de Cartan des repères, nous avons proposé l'utilisation du repère de Frenet ou du repère du transport relativement parallèle (RTRP).

Ce procédé nécessite la spécification d'un chemin reliant $\gamma_0(0)$ à $\gamma_1(0)$ et d'un chemin reliant $r_0(0)$ à $r_1(0)$ où r_0 et r_1 sont les repères R_0 et R_1 transportés en l'identité du groupe. De plus $R_0(0)$ et $R_1(0)$ doivent également être spécifiés si l'interpolation utilise le RTRP.

Lorsque X est un espace euclidien, $\gamma_0(0)$ et $\gamma_1(0)$ peuvent canoniquement être reliés par le segment qui les joint. De plus,

- la courbe interpolée est automatiquement munie d'un repère mobile du même type que celui de γ_0 et de γ_1 ,
- sa longueur totale est une fonction linéaire des longueurs de γ_0 et de γ_1 ,
- et sa courbure est majorée par une fonction monotone joignant les courbures de γ_0 et γ_1 .

Cette dernière propriété s'avèrera utile lorsque nous décrirons dans la troisième et dernière partie notre méthode de métamorphose.

Nous avons restreint notre étude à des groupes de Lie car ils sont munis d'un parallélisme canonique qui permet de comparer les hodographes de chaque courbe γ_0 et γ_1 . Nous les avons choisis connexes car il est nécessaire de relier $\gamma_0(0)$ à $\gamma_1(0)$. Enfin, afin de pouvoir intégrer l'hodographe interpolé nous avons imposé à ce groupe de Lie d'être compact ou de posséder une représentation linéaire fidèle. Une étude plus poussée des groupes de Lie sort largement du cadre de cette thèse, mais pour justifier le langage mathématique que nous avons employé, il faut s'assurer que ces restrictions sont malgré tout vérifiées par un grand nombre d'objets. Or les travaux de classification des groupes de Lie [Die75] montrent que tous les groupes de Lie connexes et commutatifs admettent une représentation fidèle (c'est également le cas des groupes compacts et simplement connexes mais la compacité suffit à nos besoins). Comme exemples classiques de groupes de Lie répondant à nos besoins citons les espaces vectoriels de dimension finie, les tores, les sphères S^1 et S^3 , tous les sous-groupes connexes de $GL(n, \mathbb{R})$ et tous les produits de ces groupes...

L'extension de ce procédé d'interpolation à des variétés riemanniennes connexes et compactes peut se faire en utilisant l'application exponentielle et le parallélisme canonique le long d'une courbe :

1. on construit un repère adapté R_0 le long de γ_0 (resp. R_1 le long de γ_1),

2. on transporte ce repère en $T_{\gamma_0(0)}X$ (resp. $T_{\gamma_1(0)}X$) en utilisant le parallélisme induit par la courbe γ_0 (resp. γ_1).
3. Puis les deux repères sont à nouveau transportés en $\gamma_t(0)$, en utilisant le parallélisme le long de l'arc joignant $\gamma_0(0)$ à $\gamma_1(0)$, pour être interpolés.
4. On en déduit un hodographe \mathbf{v}_t que l'on transporte localement en tout point en utilisant l'application exponentielle en $\gamma_t(0)$.
5. on intègre ce champ et on transporte \mathbf{v}_t en l'extrémité $\gamma_t(s_1)$ de la courbe intégrale trouvée. On transporte à nouveau ce champ en utilisant l'application exponentielle en $\gamma_t(s_1)$ et on recommence...

Ce processus a une fin : ceci peut être montré en utilisant le fait que la vitesse interpolée est majorée par les vitesses des courbes γ_0 et γ_1 et que, sur une variété compacte, le domaine de l'application exponentielle contient au moins un disque de rayon fixé (appelé le rayon d'injectivité) et non nul en tout point. En notant v_{max} un majorant de la vitesse et ϵ le rayon d'injectivité, on doit répéter au plus $\frac{v_{max}}{\epsilon}$ fois l'opération 5.

Chapitre 2

Un algorithme d'interpolation de courbes polygonales

Dans le premier chapitre nous avons défini un procédé d'interpolation de courbes dans un cadre mathématique où les objets sont de nature continue (du cardinal de \mathbb{R}). Or, l'outil principal des infographistes, l'ordinateur, ne permet pas de manipuler des objets de taille infinie, de sorte qu'au lieu de travailler sur des objets de type sous-variétés de \mathbb{R}^n , nous devons définir des *modèles* discrets pour ces objets et voir de quelle manière les propriétés connues dans \mathbb{R}^n se “transposent” sur ces modèles.

Par exemple dans [KKM91], on étudie la topologie discrète sur \mathbb{Z}^2 (et dans une moindre mesure sur \mathbb{Z}^3), on définit ce que peut être une courbe dans cet espace et on obtient ensuite sur ce modèle de courbe un équivalent du théorème de Jordan.

Dans le domaine de la topologie et de la géométrie discrète on s'intéresse d'ailleurs souvent à l'obtention, dans \mathbb{Z}^n ou dans des espaces finis, de propriétés *analogues* à celles que l'on connaît sur \mathbb{R}^n [Uni93]. Ainsi J.P. Réveillès de l'université Louis Pasteur à Strasbourg travaille sur la notion de droites discrètes dans \mathbb{Z}^2 . Ces dernières sont définies par des inéquations à coefficients entiers appelées équations diophantiennes [Rév93]. Ces droites ne possèdent pas les mêmes propriétés que les droites usuelles du plan euclidien ; en particulier elles possèdent une épaisseur. Cependant, il est possible de définir certaines notions telles que l'intersection de deux droites discrètes et d'établir un lien avec les propriétés classiques de la géométrie euclidienne. Dans ce cas précis le lien est obtenu à partir d'une construction mathématique : l'analyse non-standard [CNR92], qui permet de créer un “pont” entre l'arithmétique sur \mathbb{Z}^2 et la géométrie euclidienne sur \mathbb{R}^2 .

Nous ne nous placerons pas dans l'optique de la géométrie discrète, néanmoins nous dégagerons une notion de modèle pour faire la liaison entre les raisonnements “en continu”

et les algorithmes utilisés dans l'ordinateur. Le chapitre s'organise de la manière suivante :

- Nous présentons une notion générale de modèle après avoir rappelé les approches existantes,
- nous proposons un modèle discret de courbes,
- puis nous présentons un algorithme d'interpolation de modèles de courbe par analogie avec la méthode décrite dans le premier chapitre.

Les résultats des sections 2.2 à 2.4 ont été publiés sous forme d'un rapport INRIA [Laz95].

2.1 Du modèle continu au modèle discret

2.1.1 La notion de modèle en modélisation géométrique

Dans le domaine de l'infographie, et tout particulièrement en modélisation géométrique, les objets manipulés sont de nature géométrique et topologique et correspondent à la notion mathématique de variété différentielle. Ces objets sont couramment représentés par des polygones, des polyèdres ou des assemblages de carreaux de spline [Ris91]. Pour justifier ces choix, une réflexion sur la notion de modèle apparaît souvent en introduction de la Géométrie Solide Constructive ou CSG ([Hof89], §4.4: Representation and Model).

Dans [GV95], qui reprend [Req80], on introduit les notions *d'univers physique*, *d'univers mathématique* et *d'univers de représentation*. Appliqué à la modélisation, l'univers physique est l'ensemble des formes de \mathbb{R}^3 . Les formes sont décrites de manière implicite, par des fonctions d'appartenance. L'univers mathématique, noté M , est alors le sous ensemble des formes qui peuvent être décrites par une fonction calculable. Ceci mène les auteurs à définir M comme l'ensemble des ensembles semi-algébriques (ensembles décrits par des inéquations polynomiales). Le théorème de Stone-Weirstrass est invoqué pour justifier la puissance de description de l'univers physique par M .

Les différents processus de codage des fonctions d'appartenance des objets de M constituent autant d'univers de représentation. Un bon univers de représentation R est un univers en bijection r avec M et tel que, pour toute opération L dans M , il existe une opération L_1 dans R qui rende commutatif¹ le diagramme suivant :

$$\begin{array}{ccc} M & \xrightarrow{r} & R \\ L \downarrow & & \downarrow L_1 \\ M & \xrightarrow{r} & R \end{array}$$

1. on dit qu'un diagramme est commutatif lorsque la composition des flèches qui vont d'un objet à un autre ne dépend pas du chemin orienté suivi. Dans le diagramme ci-dessus cela signifie que $L_1 \circ r = r \circ L$.

Ce diagramme traduit une bonne adéquation entre le modèle mathématique et sa représentation, et il exprime, en particulier, qu'on peut associer à tout procédé de construction mathématique dans M un *algorithme* dans R .

Notons que dans [GV95], on se place dans le cadre d'un ordinateur capable de travailler sur les nombres réels.

En choisissant de ne s'intéresser, au départ, qu'à un certain type d'objets mathématiques représentables dans l'ordinateur (les ensembles semi-algébriques), la présentation de [GV95] a le défaut, nous semble-t-il, d'ignorer les problèmes numériques.

Nous présentons une approche originale qui, à notre avis, a le mérite de mieux mettre en valeur les questions de convergence et d'approximation.

2.1.2 Modèle et classes d'équivalence

Si nous cherchons à formuler de façon grossière le problème principal de l'infographie, nous pouvons dire qu'il s'agit de reproduire à l'aide de l'ordinateur des phénomènes physiques ou des opérations mathématiques que nous imaginons de manière continue. C'est-à-dire, en nous inspirant de la définition du mot *simulation* dans le petit Larousse, que nous cherchons à remplacer un système à étudier par un modèle (informatique) plus simple ayant un comportement analogue.

Une des difficultés réside dans la définition d'un *modèle* et d'un *comportement analogue*. Nous proposons de donner un sens plus précis à ces notions. Notons qu'il existe une théorie des modèles fondées sur la notion de *catégorie* (c.f. [Hus94]). L'emploi de cette théorie nécessite un langage et un formalisme qui nous semblent disproportionnés en comparaison de l'apport possible de ce cadre théorique. Notre objectif est en effet de définir des notions suffisamment simples pour être opérantes et suffisamment générales pour être applicables à l'ensemble des problèmes que nous avons à traiter.

Dans le domaine de la modélisation d'objets, auquel nous nous rattachons, il est souvent plus approprié d'exprimer les problèmes à résoudre en termes d'opérations mathématiques sur certaines classes d'objets, plutôt qu'en termes de simulation physique. Nous nous intéresserons donc plus spécifiquement à un univers mathématique.

Comme il est rappelé dans [GV95], cet univers ne peut être représenté de façon biunivoque dans l'ordinateur. Seules certaines classes d'objets admettent une représentation non ambiguë, c'est-à-dire peuvent être codées dans l'ordinateur de sorte que chaque objet possède un code permettant de retrouver *toute* l'information qui sert à le définir.

Nous notons \mathcal{M} l'univers des objets mathématiques et \mathcal{R} l'ensemble des objets représen-

tables². Le travail du concepteur de logiciel peut alors s'exprimer comme suit : *Comment construire un algorithme A dans \mathcal{R} qui soit analogue à une opération L dans \mathcal{M} .*

Exemple 2.1 *On peut chercher à simuler l'opération L de dérivation de l'ensemble $\mathcal{C}^p([0, 1])$, des fonctions réelles de classe \mathcal{C}^p , dans l'ensemble $\mathcal{C}^{p-1}([0, 1])$.*

Pour résoudre ce problème il faut définir des modèles informatiques pour $\mathcal{C}^p([0, 1])$ et $\mathcal{C}^{p-1}([0, 1])$, et construire un algorithme A sur ces modèles qui corresponde à L .

Comme il y a a priori moins d'objets dans \mathcal{R} que dans \mathcal{M} , il paraît naturel d'associer à un élément de \mathcal{R} un sous-ensemble, ou une classe, d'éléments de \mathcal{M} . Nous allons ainsi considérer les éléments de \mathcal{R} comme des représentants de certaines classes d'équivalences sur \mathcal{M} . Nous proposons ainsi la notion de modèle suivante :

Définition 2.1 *Soit \mathcal{C} une classe d'objets de \mathcal{M} , nous appelons **modèle** pour \mathcal{C} une suite $(\sim_n, d_n)_{n \in \mathbb{N}}$ de relations d'équivalences \sim_n sur \mathcal{C} et de distances d_n sur le quotient \mathcal{C}/\sim_n telles que*

1. *toute classe de \sim_n admet au moins un représentant dans \mathcal{R} (i.e. un représentant représentable!),*
2. *pour tous objets \mathcal{O} et \mathcal{O}' de \mathcal{M} et toute suite de représentants $(\mathcal{R}_n \mid \mathcal{R}_n \in \mathcal{C}/\sim_n)_{n \in \mathbb{N}}$ vérifiant*

$$\lim_{n \rightarrow \infty} d_n(\mathcal{O}_n, \mathcal{R}_n) = \lim_{n \rightarrow \infty} d_n(\mathcal{O}'_n, \mathcal{R}_n) = 0$$

on a $\mathcal{O} = \mathcal{O}'$ (on a respectivement noté \mathcal{O}_n et \mathcal{O}'_n les représentants de \mathcal{O} et \mathcal{O}' modulo \sim_n).

La première condition traduit la compatibilité du modèle avec le codage informatique, tandis que la seconde permet de considérer les éléments de \mathcal{C} comme des limites de suites de représentants (représentables).

Par la suite nous notons $\mathcal{O} \xrightarrow{\sim_n} \mathcal{O}_n$ la projection canonique de \mathcal{O} dans \mathcal{C}/\sim_n et $(\mathcal{O}_n)_{n \in \mathbb{N}} \xrightarrow{\text{lim}_{\sim_n}} \mathcal{O}$ le passage à la limite (lorsqu'elle existe).

Nous pouvons alors définir une notion d'algorithme relatif à une opération et des modèles.

Définition 2.2 *Soient \mathcal{C} et \mathcal{C}' deux classes de \mathcal{M} pour lesquelles nous possédons des modèles respectifs $(\sim_n, d_n)_{n \in \mathbb{N}}$ et $(\sim'_n, d'_n)_{n \in \mathbb{N}}$. Soit L une opération $\mathcal{O} \mapsto \mathcal{O}'$ de \mathcal{C} dans \mathcal{C}' , nous dirons qu'un algorithme sur \mathcal{R} est **analogue** à l'opération L relativement aux modèles $(\sim_n$*

2. Dans [GV95] ce que nous notons \mathcal{M} est appelé l'univers physique, tandis que ce que nous notons \mathcal{R} correspond à l'univers mathématique...

, $d_n)_{n \in \mathbb{N}}$ et $(\sim'_n, d'_n)_{n \in \mathbb{N}}$ si le diagramme suivant est commutatif :

$$\begin{array}{ccc} \mathcal{O} & \xrightarrow{\sim_n} & (\mathcal{O}_n)_{n \in \mathbb{N}} \\ L \downarrow & & \downarrow A \\ \mathcal{O}' & \xleftarrow{\lim_{\sim'_n}} & (A.\mathcal{O}_n)_{n \in \mathbb{N}} \end{array}$$

Ce qui exprime que \mathcal{O}' peut être obtenu soit directement par l'opération L , soit en appliquant l'algorithme A à des "modèles" \mathcal{O}_n de plus en plus précis.

Grâce à cette définition il devient possible de comparer l'efficacité de deux algorithmes A_1 et A_2 en fonction de la vitesse de convergence des suites $(A_1.\mathcal{O}_n)_{n \in \mathbb{N}}$ et $(A_2.\mathcal{O}_n)_{n \in \mathbb{N}}$.

Une classe d'objets mathématiques n'admet pas nécessairement un modèle. D'un autre côté il semble assez vain de parler d'algorithme analogue à une opération sur des ensembles qui n'admettent pas de modèles. Afin d'illustrer l'importance du modèle par rapport à l'algorithme nous détaillons ci-dessous un exemple.

Exemple 2.2 Replaçons nous dans le cadre de l'exemple 2.1 où $\mathcal{C} = \mathcal{C}^p([0, 1])$, $\mathcal{C}' = \mathcal{C}^{p-1}([0, 1])$ et L est la dérivation d'ordre 1. Nous choisissons pour \mathcal{C} le modèle suivant :

- $f \sim_n g$ si et seulement si

$$f^{(k)}\left(\frac{i}{n}\right) = g^{(k)}\left(\frac{i}{n}\right), \quad \forall i = 0, \dots, n \text{ et } \forall k = 0, \dots, m$$

où m est un entier fixé inférieur ou égal à p .

Un représentant \mathcal{R}_n d'une classe peut donc s'identifier à un $(m+1)(n+1)$ -uplet de réels $(r_i^k)_{0 \leq i \leq n, 0 \leq k \leq m}$

- nous posons alors

$$d_n(f, \mathcal{R}_n) = \sqrt{\frac{1}{n} \sum_{i,k} (f^{(k)}\left(\frac{i}{n}\right) - r_i^k)^2}$$

Ici, la validité du modèle tient au fait que :

$$\lim_{n \rightarrow \infty} d_n(f, g) = \sqrt{\sum_k \int_0^1 (f^{(k)} - g^{(k)})^2}$$

Nous choisissons un modèle du même type pour \mathcal{C}' avec un entier m' strictement inférieur à p .

• Si $m \geq 1$ et $m' < m$ nous pouvons définir simplement l'algorithme A par :

$$A.(r_i^k)_{0 \leq i \leq n, 0 \leq k \leq m} = (r_i^l)_{0 \leq i \leq n, 1 \leq l \leq m'+1}$$

Dans ce cas l'algorithme est clairement optimal.

- Si $m = m' = 0$ nous pouvons définir A par :

$$A.(r_i^0)_{0 \leq i \leq n} = (r_i'^0)_{0 \leq i \leq n}$$

avec

$$\begin{cases} r_i'^0 = n(r_{i+1}^0 - r_i^0) & \text{si } i < n \\ r_n'^0 = r_{n-1}^0 \end{cases}$$

Nous pouvons également remplacer l'expression de $r_i'^0$ par une formule du même type, telle que $\frac{n}{2}(r_{i+1}^0 - r_{i-1}^0)$. Aucune de ces méthodes n'est optimale et la vitesse de convergence dépend de l'ordre d'approximation de la dérivée.

Sur cet exemple trivial nous voyons que la richesse du modèle influe sur le choix de l'algorithme et sur son efficacité.

Notons par ailleurs que nous avons implicitement supposé l'ordinateur capable de manipuler des nombres réels. Nous pouvons cependant nous passer de cette hypothèse en remplaçant la définition de \sim_n par une relation du type³ :

$f \sim_n g$ si $f^{(k)}(\frac{i}{n})$ et $g^{(k)}(\frac{i}{n})$ ont un même développement décimal à l'ordre n .

2.1.3 Un modèle de courbe

Nous présentons un modèle discret pour les courbes définies en 1.2, c'est-à-dire les arcs compacts définis sur $[0,1]$. Ce modèle est une variante du modèle présenté dans l'exemple 2.2 précédent. Nous en donnons tout d'abord une justification plus intuitive.

Nous choisissons de représenter la topologie, l'immersion, la paramétrisation et la structure différentielle au premier ordre. Nous explicitons dans ce qui suit ce que nous entendons par la représentation de ces propriétés.

- **la topologie** : nos courbes sont définies par un ensemble de sommets, définir une topologie revient à se donner un graphe sur ces sommets. Comme nous voulons modéliser des courbes non fermées, un tel graphe correspond à la donnée d'un ordre sur les sommets.
- **l'immersion** : elle consiste ici à associer trois réels à chaque sommet (ses coordonnées dans \mathbb{R}^3), de sorte que deux sommets adjacents (ou successifs) n'aient pas les mêmes coordonnées.
- **la paramétrisation** : les $n + 1$ sommets $(\Gamma_i)_{0 \leq i \leq n}$ étant ordonnés, nous choisissons de donner le paramètre i/n au point Γ_i . En d'autres termes nous choisissons un échantillonnage uniforme de $[0,1]$.

3. Il n'est pas possible de définir une relation d'équivalence à partir d'une distance (i.e. $f \sim_n g$ si $d(f, g) < \frac{1}{n}$) car une telle relation n'est en général pas transitive.

- **La structure différentielle au premier ordre**: elle s'obtient par la donnée en chaque sommet d'un espace tangent orienté et donc par la donnée d'un vecteur unitaire de \mathbb{R}^3 .

Finalement notre modèle n'est autre qu'une courbe polygonale de \mathbb{R}^3 , dont chaque sommet est pourvu d'un vecteur unitaire.

Notons que la vitesse v_i en un point Γ_i se déduit de l'immersion en choisissant⁴ $v_i = \|\Gamma_i \Gamma_{i+1}\|$. Nous aurions pu tout aussi bien enrichir le modèle en adjoignant à chaque sommet un réel positif représentant la vitesse. À l'inverse nous avons spécifié en chaque sommet un espace tangent que l'on aurait pu déduire de l'immersion en choisissant le vecteur $\Gamma_i \Gamma_{i+1}$ normalisé pour vecteur tangent.

Nous avons fait ce choix car, en pratique, les courbes que nous échantillonnerons seront des splines et le calcul du vecteur tangent pour un paramètre donné est simple. Ceci est également vrai pour la vitesse, mais nous avons préféré limiter le nombre d'attributs du modèle pour simplifier les algorithmes. Il est clair cependant que la fidélité dans la reproduction du modèle continu est fonction de la richesse du modèle discret.

En reprenant le formalisme de la section précédente, notre modèle de courbe est ainsi déterminé par les relations et distances suivantes :

- les relations \sim_n sont déterminées par :

$$\gamma_1 \sim_n \gamma_2 \Leftrightarrow \gamma_1(\frac{i}{n}) = \gamma_2(\frac{i}{n}) \text{ et } \mathbf{t}_1(\frac{i}{n}) = \mathbf{t}_2(\frac{i}{n}) \forall i = 0, \dots, n$$

où $\mathbf{t} = \gamma' / \|\gamma'\|$.

Un représentant d'une classe peut donc s'identifier à un $(2n + 2)$ -uplet

$$\mathcal{R}_n = (\Gamma_0, \dots, \Gamma_n, \mathbf{t}_0, \dots, \mathbf{t}_n)$$

- et on définit les distances d_n par :

$$d_n(\gamma_1, \gamma_2) = \max_i \|\gamma_1(\frac{i}{n}) - \gamma_2(\frac{i}{n})\| + \max_i \|\mathbf{t}_1(\frac{i}{n}) - \mathbf{t}_2(\frac{i}{n})\|$$

De sorte que

$$\lim_{n \rightarrow \infty} d_n(\gamma_1, \gamma_2) = \|\gamma_1 - \gamma_2\|_\infty + \|\mathbf{t}_1 - \mathbf{t}_2\|_\infty$$

où

$$\|f\|_\infty = \sup_{s \in [0,1]} \|f(s)\|.$$

4. Pour le dernier sommet on peut prendre la vitesse du sommet précédent.

Ainsi une suite $(\Gamma_i^n, \mathbf{t}_i^n)_{0 \leq i \leq n}$ de représentants “converge” vers une courbe γ si et seulement si la suite de fonctions constante par morceau⁵ $\Gamma^n : s \mapsto \Gamma_{[sn]}^n$ (resp. $\mathbf{t}^n : s \mapsto \mathbf{t}_{[sn]}^n$) converge uniformément sur $[0,1]$ vers la fonction γ (resp. $\gamma'/\|\gamma'\|$).

Définition 2.3 Par la suite nous dirons qu'une suite de séquences $(f_i^n)_{0 \leq i \leq n}$ **converge** vers une fonction f si la suite de fonctions constante par morceau $f^n : s \mapsto f_{[sn]}^n$ converge uniformément vers f .

Nous pouvons désormais définir un algorithme d'interpolation de courbes discrètes qui soit analogue à la méthode L d'interpolation proposée dans le premier chapitre. C'est un algorithme A qui rend le diagramme suivant commutatif:

$$\begin{array}{ccc} \gamma_0, \gamma_1 & \xrightarrow{\sim^n} & (\Gamma_{0,i}^n, \mathbf{t}_{0,i}^n)_{n \in \mathbb{N}}, (\Gamma_{1,i}^n, \mathbf{t}_{1,i}^n)_{n \in \mathbb{N}} \\ L \downarrow & & \downarrow A \\ \gamma_t & \xleftarrow{\text{lim}\sim} & (\Gamma_{t,i}^n, \mathbf{t}_{t,i}^n)_{n \in \mathbb{N}} \end{array}$$

2.2 Algorithme d'interpolation

Nous supposons désormais données deux courbes échantillonnées $(\Gamma_{0,i}^n, \mathbf{t}_{0,i}^n)_{0 \leq i < n}$ et $(\Gamma_{1,i}^n, \mathbf{t}_{1,i}^n)_{0 \leq i < n}$ que nous désirons interpoler. Par la suite nous omettrons les indices relatifs au temps (0, 1 ou t) lorsqu'ils ne seront pas nécessaires.

2.2.1 Schéma principal

En se référant au procédé d'interpolation du premier chapitre, nous devons munir chaque courbe échantillonnée $(\Gamma_i^n, \mathbf{t}_i^n)$ d'un repère adapté (et orthonormé) \mathcal{R}_i^n . Ici, *adapté* signifie que \mathbf{t}_i^n est le premier vecteur de R_i^n .

Nous devons ensuite interpoler ce repère, la vitesse $v_i^n = \|\Gamma_i^n \Gamma_{i+1}^n\|$, et l'extrémité Γ_0^n . Cette dernière quantité peut d'ailleurs être indépendante de n , ou plus exactement, $\Gamma_{t,0}^n$ peut ne dépendre que de t .

Nous devons ensuite reconstruire une courbe avec les quantités interpolées. Cependant notre définition des courbes échantillonnées ne permet pas d'écrire:

$$\Gamma_{i+1}^n = \Gamma_i^n + \frac{1}{n} v_i^n \mathbf{t}_i^n$$

puisque dans notre modèle Γ_i^n et \mathbf{t}_i^n sont indépendants.

Afin d'être cohérent avec les données initiales, nous remplaçons cette équation par l'équation de progression suivante:

$$\Gamma_{i+1}^n = \Gamma_i^n + \frac{1}{n} R_i^n \Delta_i^n \tag{2.1}$$

5. $[x]$ désigne le plus grand entier inférieur ou égal à x

Δ_i^n est donc n fois la différence $\Gamma_{i+1}^n - \Gamma_i^n$ exprimée dans le repère R_i^n . Si $(\Gamma_i^n, \mathbf{t}_i^n)$ converge vers une courbe γ , il est facile de voir que Δ_i^n converge vers le vecteur

$$\|\gamma'\| \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = v \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}.$$

Par conséquent, en substituant l'interpolation linéaire de Δ_i^n à celle de v_i^n , nous conservons une interpolation linéaire de la vitesse et l'équation 2.1 nous permet de reconstruire la courbe interpolée, à tout instant, de façon compatible avec les données initiales.

Nous montrons en effet, en annexe B, que si \mathcal{R}_i^n et Δ_i^n convergent respectivement vers un repère mobile R et un vecteur Δ , alors la courbe (Γ_i^n) reconstruite à l'aide de l'équation 2.1 converge vers la courbe γ définie par :

$$\gamma(s) = \Gamma_0 + \int_0^s R(u)\Delta(u)du$$

où $\Gamma_0 = \Gamma_0^n$ est indépendant de n . De plus, si Δ est de la forme $v \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$, γ a pour vitesse v et R est un repère mobile adapté pour γ . L'interpolation linéaire de Δ_i^n permet effectivement de remplir cette condition (i.e. $\Delta_{t,i}^n$ converge vers $v_t \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$ avec $v_t = (1-t)v_0 + tv_1$).

Par la suite il nous reste à définir comment construire un repère adapté R_i^n à une courbe échantillonnée $(\Gamma_i^n, \mathbf{t}_i^n)$ et comment interpoler deux repères $R_{0,i}^n$ et $R_{1,i}^n$.

2.2.2 Construction d'un repère adapté à une courbe échantillonnée

Nous devons munir une courbe échantillonnée $(\Gamma_i^n, \mathbf{t}_i^n)$ d'un repère adapté R_i^n (i.e. dont le premier vecteur est \mathbf{t}_i^n). Par la suite nous identifierons les repères et les rotations de \mathbb{R}^3 , utilisant indifféremment un vocabulaire propre aux repères ou aux rotations. Par exemple le repère canonique et l'identité désignerons la même chose.

Par analogie avec le premier chapitre, nous définissons un repère de Frenet discret et un repère du transport relativement parallèle (RTRP) discret.

Le repère de Frenet discret

Le repère de Frenet discret se définit très simplement par le repère⁶

$$R_i^n = (\mathbf{t}_i^n, \mathbf{n}_i^n, \mathbf{b}_i^n)$$

avec

$$\mathbf{b}_i^n = \frac{\mathbf{t}_i^n \wedge \mathbf{t}_{i+1}^n}{\|\mathbf{t}_i^n \wedge \mathbf{t}_{i+1}^n\|} \text{ et } \mathbf{n}_i^n = \mathbf{b}_i^n \wedge \mathbf{t}_{i+1}^n.$$

De même que le repère de Frenet continu ne peut être défini que sur des courbes birégulières, sa version discrète ne peut être définie que si

$$\mathbf{t}_i^n \wedge \mathbf{t}_{i+1}^n \neq 0.$$

Dans ces conditions, il est facile de voir que si $(\Gamma_i^n, \mathbf{t}_i^n)$ converge vers une courbe γ alors R_i^n converge vers le repère de Frenet de γ .

Le RTRP discret

Nous construisons un RTRP discret à l'aide de la progression suivante :

1. nous choisissons un repère R_0^n adapté,
2. puis, par analogie avec la remarque 1.2 du premier chapitre, nous propageons ce repère de sorte que si R_i^n est le repère en Γ_i^n , R_{i+1}^n est adapté et $\|R_{i+1}^n - R_i^n\|$ est minimal pour la norme euclidienne sur les matrices.

Il est facile de voir que R_{i+1}^n est obtenu à partir de R_i^n par la rotation d'angle $\alpha_i^n = (\mathbf{t}_i^n, \widehat{\mathbf{t}_{i+1}^n})$ autour du vecteur $\mathbf{t}_i^n \wedge \mathbf{t}_{i+1}^n$ normalisé (si $\mathbf{t}_i^n \wedge \mathbf{t}_{i+1}^n$ est nul nous prenons l'identité).

Dans la section suivante nous verrons que cette construction, également introduite dans [Klo86], converge effectivement vers le repère du transport parallèle relatif par passage à la limite sur les courbes échantillonnées.

2.2.3 Interpolation des repères (première version)

Nous avons vu dans le premier chapitre que l'interpolation des repères est obtenue par l'interpolation linéaire de leur matrice de Cartan. Or dans \mathbb{R}^3 une telle matrice (antisymétrique) s'identifie à un vecteur, et réciproquement, par l'application qui associe à un vecteur \mathbf{v} la matrice $\tilde{\mathbf{v}}$ (relativement à la base canonique) de l'endomorphisme :

$$\mathbf{u} \mapsto \mathbf{v} \wedge \mathbf{u}$$

6. On pose $R_n^n = R_{n-1}^n$

La matrice de Cartan d'un repère mobile (orthonormé) est donc de la forme $\tilde{\Omega}$ où Ω est un vecteur usuellement appelé le vecteur vitesse de rotation instantanée ou VVRI. Par la suite nous exprimerons le problème de l'interpolation en terme de VVRI et non plus de matrice de Cartan. Le choix du repère de Frenet ou du RTRP correspond par exemple à l'annulation de l'une des composantes du VVRI⁷.

Dans le cas continu, le repère interpolé R_t est solution du système

$$\begin{cases} R_t'(s) = R_t(s) \tilde{\Omega}_t(s) \\ R_t(0) \text{ donné} \end{cases} \quad (2.2)$$

où

$$\Omega_t = (1 - t) \Omega_0 + t \Omega_1.$$

Afin d'obtenir une solution discrète de cette équation nous utilisons une méthode d'Euler (c.f. [Bar77]) appropriée. Nous explicitons et justifions la méthode de résolution employée dans l'annexe C. Celle-ci nous fournit une solution discrète approchée (nous supprimons l'indice t dans ce qui suit) :

$$\begin{cases} R_0^n = R(0) \\ R_{i+1}^n = R_i^n \exp\left(\frac{1}{n} \tilde{\Omega}\left(\frac{i}{n}\right)\right) \end{cases}$$

où \exp est l'application exponentielle présentée dans l'annexe C.

Dans la pratique nous utilisons une équation de propagation similaire :

$$R_{i+1}^n = R_i^n \exp\left(\frac{1}{n} \tilde{\Omega}_i^n\right) \quad (2.3)$$

où Ω_i^n converge vers Ω au sens de la définition 2.3. En combinant cette propriété de convergence et celle de la méthode d'Euler (c.f. [Bar77], p 88), il est facile de montrer que R_i^n converge vers la solution du système 2.2.

Pour compléter l'algorithme d'interpolation, il nous faut donc obtenir des approximations du VVRI pour les repères associés aux deux courbes Γ_0 et Γ_1 .

VVRI du RTRP

On peut vérifier aisément que le VVRI du RTRP est de la forme $v\kappa\mathbf{b}$ où v est la vitesse, κ la courbure et \mathbf{b} la binormale à la courbe *exprimée dans le repère local*. Notons que ce produit est toujours défini pour une courbe régulière de classe C^2 car là où la (bi)normale n'est pas définie, la courbure est nulle.

Par ailleurs on vérifie assez simplement que :

$$\lim_{u \rightarrow 0} \frac{\arccos\langle \mathbf{t}(s), \mathbf{t}(s+u) \rangle}{u} = v(s)\kappa(s)$$

7. En annulant la 3^{ème} composante on retombe sur le repère de Frenet tourné de $\pm \frac{\pi}{2}$.

Ce qui montre que si α_i^n est l'angle entre les tangentes \mathbf{t}_i^n et \mathbf{t}_{i+1}^n (voir figure 2.1), $n\alpha_i^n$ converge vers $v(s)\kappa(s)$. On en déduit que

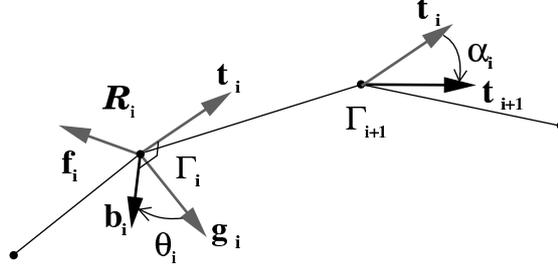


FIG. 2.1 – les angles α_i^n et θ_i^n (les indices n sont supprimés sur la figure).

$$\Omega_i^n = n\alpha_i^n \tilde{\mathbf{b}}_i^n \quad (2.4)$$

fournit une bonne approximation du VVRI. On a par conséquent, d'après l'équation 2.3 :

$$R_{i+1}^n = R_i^n \exp(\alpha_i^n \tilde{\mathbf{b}}_i^n). \quad (2.5)$$

Compte tenu des propriétés de l'exponentielle, cela signifie que R_{i+1}^n se déduit de R_i^n , comme annoncé dans la section précédente, par rotation autour de la binormale de l'angle α_i^n entre les deux tangentes \mathbf{t}_i^n et \mathbf{t}_{i+1}^n . On retrouve ainsi les résultats de [Klo86] sans aucun calcul.

Par ailleurs, le VVRI du RTRP s'exprime dans le repère local sous la forme :

$$\Omega = v\kappa \begin{pmatrix} 0 \\ -\sin \theta \\ \cos \theta \end{pmatrix}$$

où θ est une primitive de la torsion τ (c.f. [Bis75]). Si on note $(\mathbf{t}_i^n, \mathbf{f}_i^n, \mathbf{g}_i^n)$ les vecteurs du repère R_i^n , une approximation θ_i^n de l'angle θ s'obtient simplement comme on le voit sur la figure 2.1 :

$$\theta_i^n = (\mathbf{g}_i^n, \widehat{\mathbf{b}}_i^n)$$

d'où

$$\Omega_i^n = n\alpha_i^n \begin{pmatrix} 0 \\ -\sin \theta_i^n \\ \cos \theta_i^n \end{pmatrix}.$$

Nous voyons ainsi qu'il est possible d'interpoler linéairement le produit $v\kappa$ (resp. $v\tau$) de la vitesse par la courbure (resp. par la torsion) en interpolant linéairement les angles $\alpha_{t,i}^n$ et $\theta_{t,i}^n$. Nous obtenons ainsi la même interpolation (de l'arc géométrique) qu'avec le repère de Frenet.

En munissant (Γ_i^n) du RTRP discret nous avons donc deux possibilités :

1. ou bien interpoler linéairement le VVRI Ω_i^n ,
2. ou bien interpoler linéairement les angles $\alpha_{t,i}^n$ et $\theta_{t,i}^n$.

Dans les deux cas, la courbe interpolée est toujours munie du RTRP, ce qui peut être un avantage compte tenu de la propriété de minimisation de la rotation du RTRP.

VVRI du repère de Frenet

Le VVRI du repère de Frenet est de la forme $v(-\kappa\mathbf{n} + \tau\mathbf{b})$ où \mathbf{n} est la normale à la courbe et τ sa torsion. Cependant, sa version discrète

$$\Omega_i^n = -n\alpha_i^n \tilde{\mathbf{n}}_i^n + n\beta_i^n \tilde{\mathbf{b}}_i^n$$

où $\beta_i^n = (\mathbf{b}_i^n, \widehat{\mathbf{b}}_{i+1}^n)$, ne permet pas de retrouver le repère de Frenet comme il a été défini dans la section précédente. Il faut ainsi calculer le VVRI discret indirectement à partir des matrices $R_i^{-1} R_{i+1}$ obtenues par construction sur les courbes discrètes initiales. Plus précisément on a

$$\tilde{\Omega}_i^n = \log((R_i^n)^{-1} R_{i+1}^n)$$

log étant l'inverse de exp au voisinage de l'identité.

Étant donné le caractère intrinsèque du procédé d'interpolation que l'on cherche à simuler, on peut se demander s'il n'existe pas un calcul plus direct de l'interpolation des repères qui ne fasse pas intervenir le VVRI. Dans la section suivante nous montrons en effet qu'un calcul simple permet de se passer du calcul du VVRI.

2.2.4 Interpolation des repères (seconde version)

La relation 2.3 nous incite à ajouter à l'équation 2.1 de propagation des points une équation de propagation des repères :

$$R_{i+1}^n = R_i^n r_i^n \tag{2.6}$$

On obtient ainsi deux suites $(r_{0,i}^n)_{0 \leq i \leq n}$ et $(r_{1,i}^n)_{0 \leq i \leq n}$ de rotations correspondant aux repères adaptés $(R_{0,i}^n)_{0 \leq i \leq n}$ et $(R_{1,i}^n)_{0 \leq i \leq n}$. Lorsque n est assez grand, $r_{0,i}^n$ et $r_{1,i}^n$ sont proches de

l'identité. On peut par conséquent appliquer l'interpolation locale évoquée dans la section 1.1. Plus précisément, on peut définir $r_{t,i}^n$ comme le point de paramètre t sur la géodésique (de $SO(3, \mathbb{R})$) joignant $r_{0,i}^n$ et $r_{1,i}^n$.

On en déduit progressivement $R_{t,i}^n$ à l'aide de 2.6 et de la rotation $R_{t,0}^n = R_{t,0}$ obtenue à partir d'un chemin qui relie $R_{0,0}$ à $R_{1,0}$. Dans la pratique ce chemin est calculé à l'aide d'une (de la) géodésique les joignant.

En annexe D on montre que ce schéma correspond effectivement à l'interpolation linéaire des VVRI. Cela tient en partie au fait que $n(R_i^n)^{-1}(R_{i+1}^n - R_i^n) = n(r_i^n - Id)$ converge vers $\tilde{\Omega}$ si R_i^n converge vers une solution de $R' = R\tilde{\Omega}$.

Calcul avec les quaternions

L'usage des quaternions pour représenter les rotations est très répandu en infographie. Comme il est rappelé dans [BCGH92, HP93], les quaternions présentent l'avantage d'être définis par quatre coordonnées au lieu des neuf coefficients d'une rotation (autrement dit les quaternions sont naturellement plongés dans \mathbb{R}^4 tandis que les rotations le sont dans \mathbb{R}^9).

Cet usage se justifie par l'existence d'un revêtement simple (à deux feuilletés) des rotations par les quaternions unitaires (i.e. de norme égale à 1). Ce revêtement est de plus une homothétie locale de rapport 2 : les géodésiques autour d'un quaternion sont envoyées sur les géodésiques autour de la rotation correspondante mais de longueur double. Notons que le terme géodésique peut se référer ici aussi bien à la structure de groupe de Lie de ces ensembles qu'à leur structure de sous-variété riemannienne usuelle (c.f. également l'annexe C à ce sujet).

Ce revêtement associe au quaternion unitaire $(w, x, y, z) = (w, \mathbf{V})$ la rotation :

$$r = Id + 2w\tilde{\mathbf{V}} + 2\tilde{\mathbf{V}}^2$$

Inversement, la rotation d'angle α autour du vecteur \mathbf{b} correspond à l'un des deux quaternions

$$\pm (\cos(\alpha/2), \sin(\alpha/2) \mathbf{b})$$

Ainsi, en choisissant deux quaternions $q_{0,i}^n$ et $q_{1,i}^n$ correspondant aux deux rotations $r_{0,i}^n$ et $r_{1,i}^n$, on peut effectuer le calcul de $r_{t,i}^n$ en terme de quaternion (on prendra soin de choisir $q_{0,i}^n$ et $q_{1,i}^n$ "proches", c'est-à-dire tels que leur produit scalaire dans \mathbb{R}^4 soit positif). On a alors

$$q_{t,i}^n = \text{slerp}(q_{0,i}^n, q_{1,i}^n, t)$$

où *slerp* est la procédure d'interpolation des quaternions définie dans [Sho87].

Quaternions et RTRP : le calcul avec les quaternions est particulièrement simple dans le cas du RTRP puisque l'angle et l'axe des rotations r_i^n est connu à l'avance. On a ainsi :

$$q_i^n = \pm (\cos(\alpha_i^n/2), \sin(\alpha_i^n/2) \mathbf{b}_i^n)$$

où α_i^n et \mathbf{b}_i^n sont définis comme précédemment.

2.3 Extension

On peut reprocher aux modes d'interpolation précédents d'être trop contraints et de ne pas laisser assez de possibilités à l'utilisateur. Afin de donner plus de liberté à l'utilisateur nous proposons de modifier l'interpolation par l'intermédiaire d'une torsion supplémentaire et d'une interaction sur la vitesse locale de transformation. Ces modifications sont obtenues par la spécification de deux champs scalaires le long de chaque courbe à interpoler.

2.3.1 Torsion des repères

Dans notre implémentation, l'utilisateur a la possibilité d'associer interactivement des valeurs réelles en certains points des courbes. Ces valeurs sont ensuite interpolées en chaque point des courbes (au moyen d'une spline) et interprétées comme des angles. Chaque repère est ensuite tourné autour de la tangente en utilisant ces angles. Ainsi le repère R adapté à une courbe est remplacé par

$$\bar{\mathcal{R}} = \mathcal{R} r_\varphi$$

où r_φ est la rotation d'angle φ autour de l'axe des x . Le VVRI $\bar{\Omega}$ de $\bar{\mathcal{R}}$ est relié au VVRI Ω de R :

$$\bar{\Omega} = r_{-\varphi} \Omega - \varphi' \vec{t}$$

Pour interpoler deux repères "tordus" $\bar{\mathcal{R}}_0$ et $\bar{\mathcal{R}}_1$ nous pouvons soit interpoler linéairement les quantités Ω et φ soit interpoler linéairement $\bar{\Omega}$. Dans le premier cas la géométrie de la courbe interpolée n'est pas modifiée mais possède une "torsion interne" qui peut être utile si les repères servent par ailleurs de système de coordonnées comme nous le verrons dans la dernière partie de ce manuscrit. La figure 2.2 montre l'effet de la torsion sur une interpolation suivant que l'on interpole séparément ou non Ω et φ (la torsion varie de 0 à 4π le long de la courbe initiale). Afin de visualiser la torsion des repères nous avons habillé les courbes d'un cylindre dont les sections carrées sont attachées à chaque repère.

Notons que cette torsion supplémentaire permet de munir une courbe de n'importe quel repère adapté, puisque deux repères adaptés se déduisent l'un de l'autre par rotation autour de la tangente.

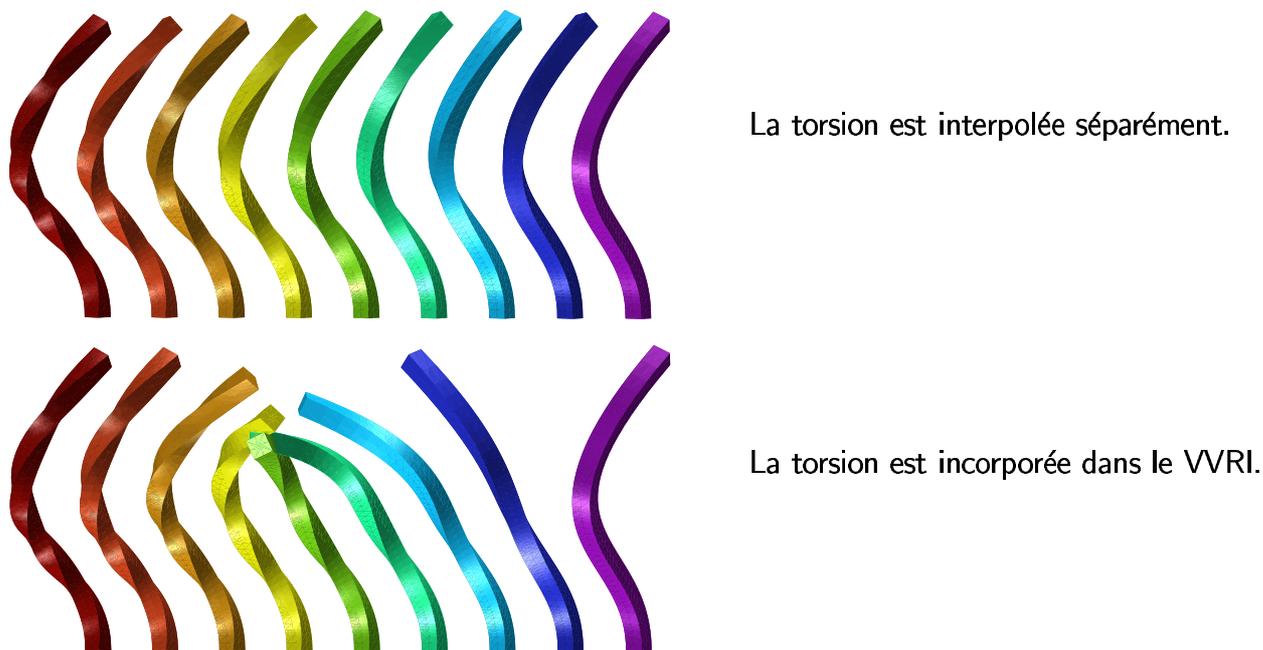


FIG. 2.2 – Influence d'une torsion supplémentaire.

2.3.2 Contrôle de la vitesse de transformation

Le second champ scalaire est utilisé pour contrôler la vitesse d'interpolation. Plus précisément nous remplaçons le temps t dans les formules d'interpolation linéaire en un point quelconque par une fonction de t et de la valeur du champ scalaire au point considéré. De sorte qu'une quantité x_i sera interpolée par une formule du type :

$$x_{t,i} = (1 - f(t, i)) x_{0,i} + f(t, i) x_{1,i}$$

f vérifiant :

$$f(0, i) = 0 \text{ et } f(1, i) = 1$$

Un exemple de modification de la vitesse d'interpolation est illustré figure 2.3.

2.3.3 Redressement automatique

Le mode d'interpolation défini dans la première partie permet également de redresser automatiquement une courbe, c'est-à-dire de transformer (à longueur constante) une courbe en une géodésique. Il suffit pour cela de rétracter le repère R_0 sur l'identité. $R_t(s)$ est donc le point de paramètre t sur la géodésique (dans $SO(3, \mathbb{R})$) qui joint $R_0(s)$ à l'identité (c.f.

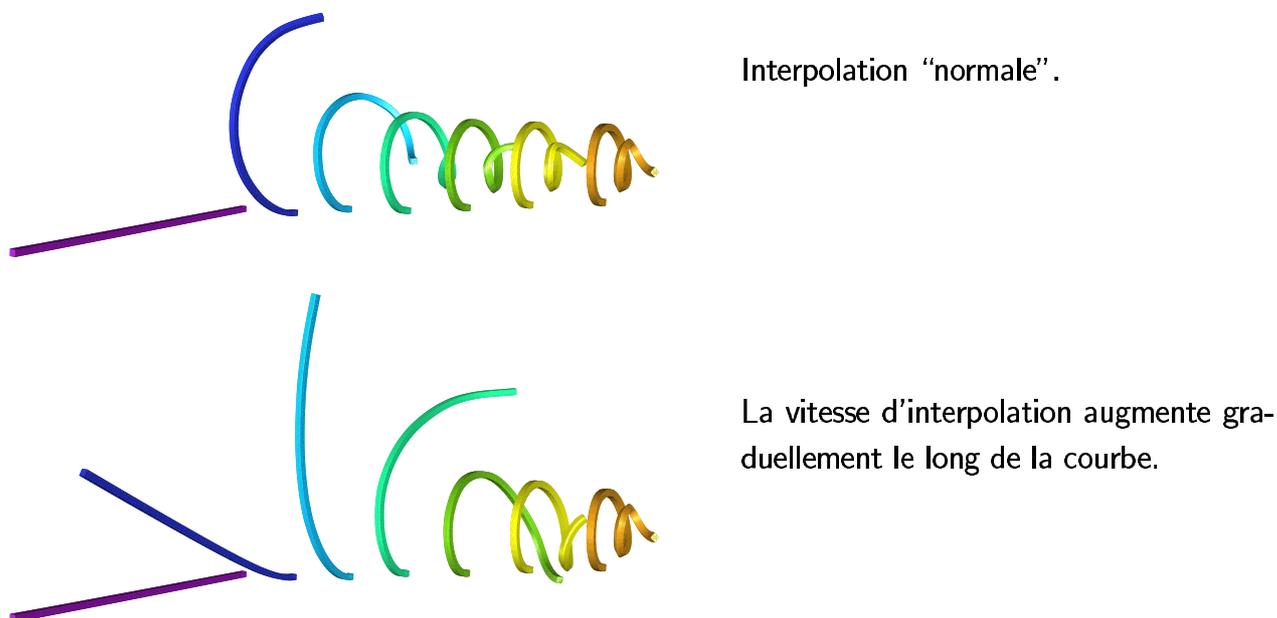


FIG. 2.3 – Influence du contrôle de la vitesse d’interpolation. Les courbes interpolées sont visualisées aux temps $t = 0, 0.1, 0.2, 0.35, 0.5, 0.75, 1$.

figure 2.4 gauche). Sur le plan algorithmique cela correspond, en se référant à la seconde méthode d’interpolation, à calculer $r_{t,i}^n$ par :

$$r_{t,i}^n = \text{slerp}(r_{0,i}^n, Id, t)$$

Un exemple de redressement est illustré sur la figure 2.4.

2.4 Implémentation et résultats

Dans notre implémentation nous n’avons utilisé que le RTRP en raison de sa propriété de “minimisation de la torsion”. Nous avons ainsi deux modes d’interpolation , mis à part les paramètres supplémentaires de la section 2.3 :

1. L’un utilise l’interpolation à l’aide des quaternions (c.f. section 2.2.4). Nous appelons Q cette méthode par référence à son mode de calcul.
2. l’autre utilise l’interpolation linéaire des angles α_i et θ_i (c.f. la fin du §VVRI du RTRP, de la section 2.2.3). Nous appelons A cette méthode par référence à son mode de calcul.

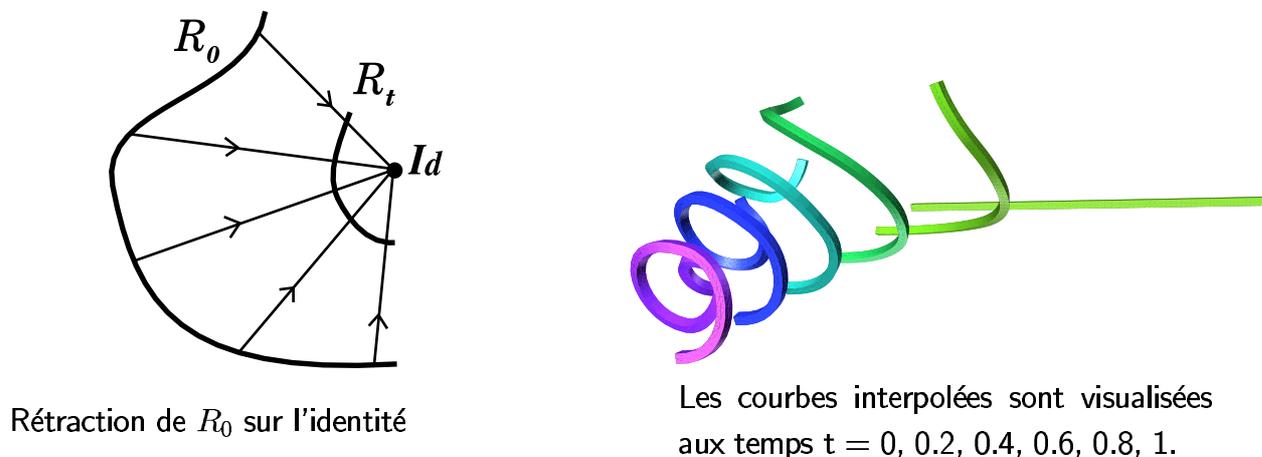


FIG. 2.4 – Redressement automatique d'une courbe

Nous résumons ci-dessous les étapes de calcul de l'interpolation de deux courbes échantillonnées $(\Gamma_{0,i}, \mathbf{t}_{0,i})$ et $(\Gamma_{1,i}, \mathbf{t}_{1,i})$ pour chacune des méthodes Q et A.

Calcul de l'interpolation par la méthode Q :

initialisation

Pour chaque courbe échantillonnée :

1. calculer les angles $\alpha_i = \widehat{(\mathbf{t}_i, \mathbf{t}_{i+1})}$, les binormales $\mathbf{b}_i = (\mathbf{t}_i \wedge \mathbf{t}_{i+1}) / \|\mathbf{t}_i \wedge \mathbf{t}_{i+1}\|$ et les quaternions $q_i = (\cos(\alpha_i/2), \sin(\alpha_i/2)\mathbf{b}_i)$,
2. choisir un repère $\mathcal{R}_0 = (\mathbf{t}_0, \mathbf{f}_0, \mathbf{g}_0)$ et le propager à l'aide de l'équation 2.5,
3. calculer les coordonnées locales $\Delta_i = \mathcal{R}_i^{-1}(\Gamma_{i+1} - \Gamma_i) = \mathcal{R}_i^\top(\Gamma_{i+1} - \Gamma_i)$

Puis on obtient la courbe interpolée à chaque instant t :

interpolation à l'instant t

1. interpoler $(\Gamma_{t,0}, \mathcal{R}_{t,0})$ à partir de $(\Gamma_{0,0}, \mathcal{R}_{0,0}), (\Gamma_{1,0}, \mathcal{R}_{1,0})$ et t (on peut utiliser les quaternions),
2. calculer pour chaque i : $q_{t,i} = \text{slerp}(q_{0,i}, q_{1,i}, t)$ et $\Delta_{t,i} = \text{lerp}(\Delta_{0,i}, \Delta_{1,i}, t)$,
3. reconstruire les $\Gamma_{t,i}$ à l'aide des équations 2.6 (convertie en terme de quaternion) et 2.1

Calcul de l'interpolation par la méthode A :

initialisation

Pour chaque courbe échantillonnée nous reprenons l'initialisation de Q et ajoutons le calcul des angles $\theta_i = (\widehat{\mathbf{g}_i, \mathbf{b}_i})$ dans l'étape 3 :

1. calculer les angles $\alpha_i = (\widehat{\mathbf{t}_i, \mathbf{t}_{i+1}})$, les binormales $\mathbf{b}_i = (\mathbf{t}_i \wedge \mathbf{t}_{i+1}) / \|\mathbf{t}_i \wedge \mathbf{t}_{i+1}\|$ et les quaternions $q_i = (\cos(\alpha_i/2), \sin(\alpha_i/2)\mathbf{b}_i)$,
2. choisir un repère $\mathcal{R}_0 = (\mathbf{t}_0, \mathbf{f}_0, \mathbf{g}_0)$ et le propager à l'aide de l'équation 2.5,
3. calculer les coordonnées locales $\Delta_i = \mathcal{R}_i^{-1}(\Gamma_{i+1} - \Gamma_i) = \mathcal{R}_i^\top(\Gamma_{i+1} - \Gamma_i)$ et les angles $\theta_i = (\widehat{\mathbf{g}_i, \mathbf{b}_i})$

interpolation à l'instant t

1. reprendre l'étape 1 de la méthode Q,
2. interpoler linéairement les quantités $\alpha_{t,i}, \theta_{t,i}$ et $\Delta_{t,i}$,
3. reconstruire les $\Gamma_{t,i}$ à l'aide de ces données en utilisant les équations 2.4, 2.3 et 2.1.

Nous présentons quelques interpolations de courbes sur les figures 2.5 et 2.6. Les courbes sont obtenues par échantillonnage de splines cubiques et nous avons habillés les courbes d'un cylindre à base carrée afin de visualiser la torsion des repères. Les mêmes courbes initiales et finales sont utilisées dans les figures 2.5 et 2.6. La figure 2.5 montre une interpolation avec la méthode A, vue de face et de dessus. La figure 2.6 montre une interpolation avec la méthode Q vu de face pour deux niveaux d'échantillonnage différents. La courbe sous-jacente à la figure du dessus comporte 40 points de discrétisation, tandis que celle sous-jacente à la figure du dessous en comporte 10. La comparaison de ces deux images montre la très bonne stabilité de notre algorithme.

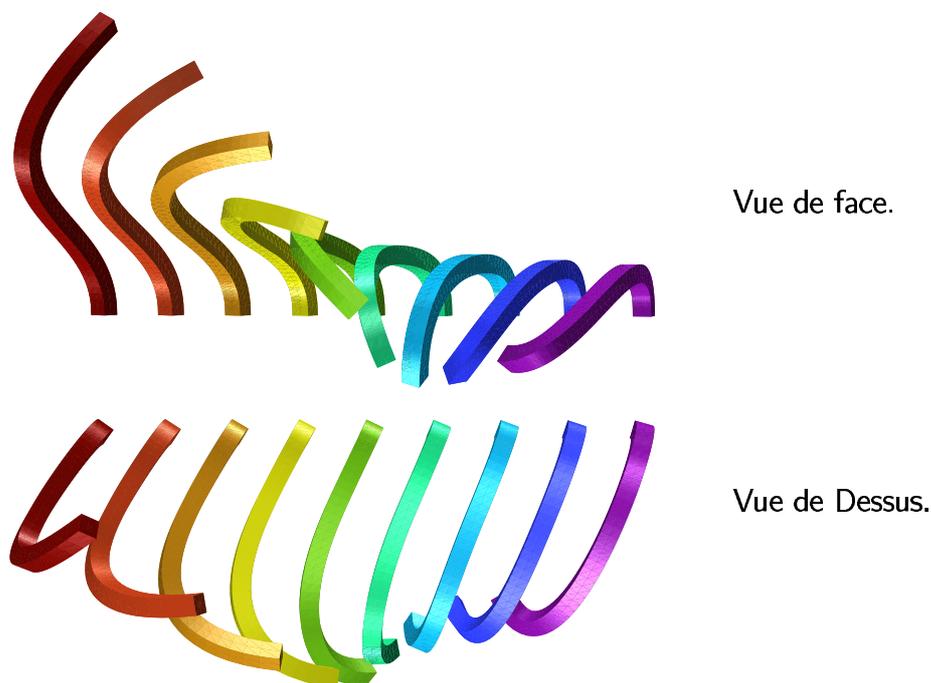


FIG. 2.5 – Interpolation par la méthode A.

2.5 Conclusion

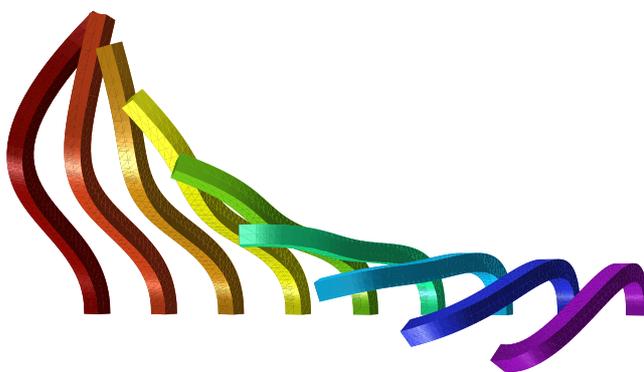
Nous avons présenté une nouvelle technique d'interpolation de courbes polygonales tridimensionnelles. Pour cela nous nous sommes appuyés sur un raisonnement dans le “continu” (c.f. chapitre 1) et nous avons établi avec précision une *analogie* avec un modèle combinatoire. Cette analogie utilise une notion originale de *modèle* décrite dans la section 2.1.

Les propriétés du procédé continu font que notre méthode est particulièrement bien adaptée pour le contrôle local (des courbures) et global (des longueurs) de la géométrie des courbes interpolées.

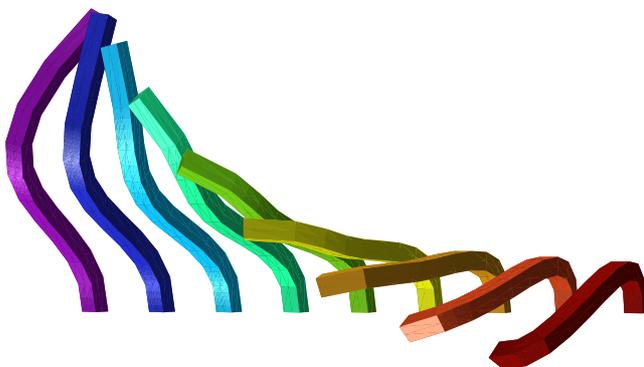
Par ailleurs, notre souci de définir dans le cas continu des procédés les plus intrinsèques possibles nous a permis d'obtenir des algorithmes les plus simples possibles et nécessitant le moins de paramètres possibles.

Afin d'obtenir une plus grande variété de transformations, nous offrons à l'utilisateur la possibilité de contrôler la torsion et la vitesse de transformation des courbes.

Notons enfin que l'extrême simplicité des algorithmes résultants permet d'obtenir des animations en temps interactif.



Vue de face. Les courbes ont 40 points.



Les courbes ont 10 points.

FIG. 2.6 – Interpolation par la méthode Q.

Deuxième partie

Aide à la création d'axe

Comme nous l'avons précisé dans l'introduction générale, les métamorphoses que nous présentons dans la dernière partie utilisent un axe défini dans chaque objet à métamorphoser. Nous avons montré comment animer ces axes, une des étapes nécessaires de la métamorphose. Dans cette partie nous étudions un moyen pour construire automatiquement ces axes dans les objets, c'est à dire un moyen d'associer une courbe squelette à un objet de forme allongée.

Chaque axe correspondant à un objet doit permettre de le paramétrer. Les contraintes permettant de définir un axe dépendent donc du mode de paramétrisation choisi. Dans notre cas les objets sont paramétrés de la manière suivante : une courbe γ permet de paramétrer un objet si celui-ci peut se décomposer en trois parties, une partie cylindrique fermée par deux calottes hémisphériques. Les calottes doivent respectivement être étoilées par rapport aux deux extrémités de la courbe et la partie cylindrique doit admettre une paramétrisation de la forme :

$$(s, \theta) \mapsto \gamma(s) + r(s, \theta) (\cos \theta \mathbf{e}_2(s) + \sin \theta \mathbf{e}_3(s))$$

où $(\mathbf{e}_2, \mathbf{e}_3)$ est un repère normal à γ et r une fonction positive.

L'existence d'une telle décomposition n'est assurée que pour un certain type d'objet que nous appelons **objet étoilé par rapport à l'axe γ** .

La difficulté pour montrer qu'un objet est étoilé par rapport à un axe et pour construire automatiquement un tel axe semble indiquer que cette hypothèse d'étoilement est très restrictive. Néanmoins elle permet de définir des métamorphoses très simplement comme nous le verrons dans la dernière partie, et constitue de plus une extension par rapport à l'approche de [KPC91] qui suppose l'objet étoilé par rapport à un point. C'est pourquoi nous avons conservé ce type d'objet.

Dans la pratique nous n'avons pas réussi à formaliser un algorithme qui réponde à ces deux questions :

1. dire si un objet peut être étoilé par rapport à un axe,
2. dans l'affirmative, proposer un tel axe.

Par contre nous proposerons un moyen de fournir une courbe pour un objet quelconque qui pourra servir de point de départ à un utilisateur. Ceci explique le titre de cette partie qui se restreint à une *aide* à la création d'axe.

Par ailleurs nous avons jugé l'approche que nous avons développée suffisamment intéressante en elle-même pour la présenter ici.

Cette partie s'organise de la manière suivante :

- nous présentons d'abord la notion d'*axe médian* abondamment utilisé en analyse d'image et en reconnaissance de formes et nous montrons en quoi cet axe médian n'est pas adapté à notre besoin,

- puis nous exposons, dans un esprit analogue au premier chapitre de la partie I, les principes qui guident notre approche,
- nous développons ensuite un algorithme de décomposition d'un objet polyédrique en bandelettes cylindriques à partir duquel nous obtenons un axe,
- nous concluons sur des voies possibles de recherche pour obtenir un axe qui réponde à la condition d'étoilement.

Chapitre 1

Squelettisation et axe médian

La représentation des objets à partir de squelettes a largement été utilisée en analyse d'image pour la reconnaissance de forme. Un squelette permet en effet de réduire l'information géométrique ou topologique d'un objet et il est plus simple de tester la ressemblance des squelettes d'objets que celle des objets eux-mêmes.

Dans une certaine mesure il est même possible de parler de codage puisqu'à partir d'un squelette et d'une fonction positive définie sur ce squelette on peut reconstruire une forme. Des algorithmes ont été proposés pour des formes simples, en particulier en CAO [GD95].

Comme il est décrit dans l'état de l'art introductif de [MR95] il existe plusieurs façons de définir le squelette d'un objet, également appelé *axe médian*. Voici les principales définitions que l'on trouve dans la littérature :

1. c'est le centre des boules maximales incluses dans l'objet,
2. c'est le lieu des points de rencontre des fronts d'une onde sphérique se propageant à vitesse constante à partir de la frontière de l'objet (analogie avec le feu de prairie),
3. c'est l'ensemble des points crêtes (ridge points) pour l'application qui à un point intérieur à l'objet associe sa distance au complémentaire de l'objet.

Ces définitions sont valides en toutes dimensions et fournissent des ensembles squelettes sensiblement équivalents (des précautions doivent cependant être prises concernant la régularité de l'objet et de sa frontière). Chaque définition donne lieu à des algorithmes de construction spécifiques. Ainsi la première mène à la construction du diagramme de Voronoï pour des ensembles de points qui approximent la surface de l'objet (c.f. [Sch89]). La seconde conduit à des opérations d'amincissement sur des images discrètes, tandis que la dernière induit le recours à des méthodes de recherches de courbes extrémales dans un champ de potentiel à l'aide de modèles de courbes déformables (snakes) (c.f. [LL89]).

L'étude des propriétés du squelette est assez délicate. Elle a été en partie menée à l'aide des outils de la *morphologie mathématique* ([Ser82, Mat88]).

Comme l'étude précise et préalable de ce que l'on cherche à construire est de première importance nous décrirons dans ce qui suit les grandes lignes de l'approche proposée en morphologie mathématique.

1.1 Axe médian et morphologie mathématique

Nous résumons ici, à partir de [Ser82, Pre], les notions topologiques de bases utilisées en morphologie mathématique et exposons brièvement les principaux résultats de [Mat88].

Ces résultats portent sur la connexité du squelette et la semi-continuité de l'application qui associe son squelette à un objet. Le terme de semi-continuité n'a de sens que si nous définissons les ensembles source et but de l'application squelette et une topologie sur ces ensembles.

Les objets que nous étudions sont des sous-ensembles d'un espace topologique E (dans la pratique \mathbb{R}^n). Il est pratique pour des études théoriques de considérer des classes d'objets particulières liées à la topologie de E . On peut ainsi s'intéresser aux sous-ensembles fermés, ouverts, ou encore compacts. Chacune de ces classes d'objets peut être munie d'une topologie en relation avec celle de E :

- si l'on considère l'ensemble noté \mathcal{F} des fermés, en supposant que E est localement compact (i.e. tout point admet un voisinage compact) et séparable (i.e. E admet une base dénombrable d'ouverts), on peut définir une topologie sur \mathcal{F} appelée *topologie en tout ou rien* que nous explicitons en appendice,
- si l'on considère l'ensemble noté \mathcal{G} des ouverts, on peut se placer dans les conditions précédentes et remarquer que l'application "complémentaire", qui passe au complémentaire dans E , réalise une bijection de \mathcal{G} sur \mathcal{F} . La topologie sur \mathcal{G} est alors l'image réciproque de celle de \mathcal{F} par cette application ou encore la topologie la moins fine sur \mathcal{G} qui la rende continue. Cette remarque simple pourra intéresser le lecteur de [Ser82] ou [Mat88] où la topologie sur \mathcal{G} est abondamment utilisée sans jamais être définie (sic).
- si l'on considère l'ensemble des compacts non vides, il est commode de supposer que E est un espace métrique. Dans ce cas on peut définir une distance sur l'ensemble des compacts, la distance de Hausdorff. Dans [Sch89] on utilise en fait une autre topologie (sur *tous* les compacts, vides ou non), la *topologie myope*, dont la définition est proche de celle sur \mathcal{F} (mais plus fine que celle induite par \mathcal{F}).

Pour ne pas alourdir le texte nous donnons en appendice E des précisions sur les notions de convergence, limite et semi-continuité pour la topologie en tout ou rien.

Deux résultats sur le squelette Lorsque $E = \mathbb{R}^n$ on définit le squelette d'un objet comme l'adhérence des centres des boules maximales incluses dans l'objet (une boule est maximale si elle n'est strictement incluse dans aucune autre boule elle-même incluse dans l'objet).

Dans [Mat88] on étudie la continuité de l'application squelette. Suivant qu'on définit cette application sur \mathcal{F} ou \mathcal{G} ou que l'on considère des boules ouvertes ou fermées on obtient des résultats différents. Ainsi l'application squelette définie sur \mathcal{F} à partir de boules fermées n'est continue ni supérieurement ni inférieurement comme le montre l'exemple de la figure 1.1 extrait de [Mat88]. Par contre l'application squelette définie sur \mathcal{G} à partir de boules

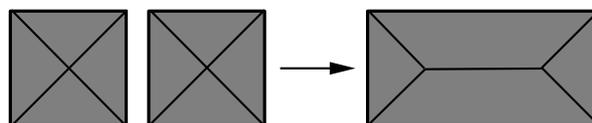


FIG. 1.1 – La réunion d'un carré C unitaire et de son translaté $C_{1+1/n}$, le long d'une de ces arêtes, converge dans \mathcal{F} vers un rectangle lorsque la translation de longueur $1 + 1/n$ tend vers 1. Cependant la limite supérieure du squelette de ce $C \cap C_{1+1/n}$ n'est pas contenue dans le squelette du rectangle, et la limite inférieure ne contient pas le squelette du rectangle.

ouvertes est semi-continue inférieurement (de \mathcal{G} dans \mathcal{F}). La démonstration utilise la caractérisation (3) (de l'appendice E) de la semi-continuité inférieure et le fait que pour toute suite convergente $G_i \mapsto G$ dans \mathcal{G} et toute boule maximale B dans G , il existe une suite de boules maximales dans les G_i qui converge vers B (dans \mathcal{G}).

Le second résultat porte sur la connexité du squelette et stipule que si G est un ouvert connexe ne contenant aucun demi-espace (i.e. de boule de rayon infini) alors le squelette est connexe.

L'étude de la continuité de l'application squelette permet de mieux prévoir le comportement des algorithmes correspondants et en particulier leur stabilité par rapport aux données numériques. Cependant le résultat précédent de semi-continuité paraît assez difficile à interpréter d'autant plus que les espaces \mathcal{F} et \mathcal{G} ne sont pas métrisables.

1.1.1 Algorithmes de construction de l'axe médian

Il existe deux grandes familles d'algorithmes de construction du squelette d'un objet. L'une utilise une formulation à partir des opérations de base de la morphologie mathématique

que sont l'érosion et la dilatation (c.f. [Ser82]), l'autre relie le squelette à la triangulation de Delaunay d'une discrétisation de la frontière de l'objet.

La première formulation se prête bien à des calculs dans des espaces discrets \mathbb{Z}^2 (resp. \mathbb{Z}^3). L'idée est d'amincir l'objet progressivement pixel par pixel (resp. voxel par voxel) jusqu'à atteindre le squelette de l'objet (voir par exemple [HP84]).

La seconde utilise le fait que les sphères de Delaunay¹ associées à une triangulation ne contiennent aucun sommet de l'ensemble triangulé en leur intérieur. Quand l'échantillonnage de la frontière est très fin cela signifie intuitivement que les sphères de Delaunay sont tangentes à la frontière de l'objet en plusieurs points et sont donc maximales.

Ainsi M. Schmitt montre dans [Sch89] que pour tout objet compact dont la frontière est une réunion d'hypersurfaces de classe \mathcal{C}^3 : *l'ensemble des centres des sphères de Delaunay associées à toute suite d'échantillonnage ϵ -dense de la frontière de l'objet² converge au sens de la topologie en tout ou rien (et au sens de la topologie myope) vers le squelette de l'objet quand ϵ tend vers 0*. Dans le cas d'objets plans à frontière assez régulière le squelette est un graphe planaire localement fini. M. Schmitt relie la topologie du diagramme de Voronoï d'une discrétisation de la frontière avec celle du squelette de l'objet. Dans le cas tridimensionnel nous n'avons pas de résultat mais il semble que pour des objets suffisamment réguliers le squelette soit une union finie de variétés à bord de dimension 1 ou 2.

Il existe néanmoins de nombreux algorithmes pour calculer le squelette à partir du diagramme de Voronoï aussi bien en 2D [MR95, Cho95] (dans ces articles on détermine le diagramme de Voronoï de polygones avec des côtés rectilignes ou non, et pas simplement de leurs sommets) qu'en 3D [GYKD91].

1.2 Conclusion

Si l'axe médian se prête bien à des traductions algorithmiques, il ne correspond cependant pas à ce que nous recherchons, à savoir une courbe (c.f figure 1.2). On peut alors penser à extraire une courbe intéressante à partir de l'axe médian. La première idée qui vient à l'esprit est de prendre le squelette du squelette. Cependant la tâche est ici un peu plus ardue si l'on veut utiliser un algorithme de type Delaunay, car les surfaces à trianguler (correspondant aux différentes branches du squelette) peuvent être gauches. Pour bien faire il faudrait utiliser une métrique induite sur chaque morceau du squelette. En fait, en supposant que les courbures des morceaux sont grandes par rapport à leur degré d'échantillonnage, on peut utiliser une heuristique comme dans [ST92] : partant de 3 sommets, on regarde si la plus petite sphère

1. Pour un état de l'art sur les diagrammes de Voronoï et les triangulations de Delaunay voir [Aur90]

2. Un échantillonnage est dit ϵ -dense si toute boule de rayon ϵ centrée sur la frontière de l'objet contient au moins un échantillon

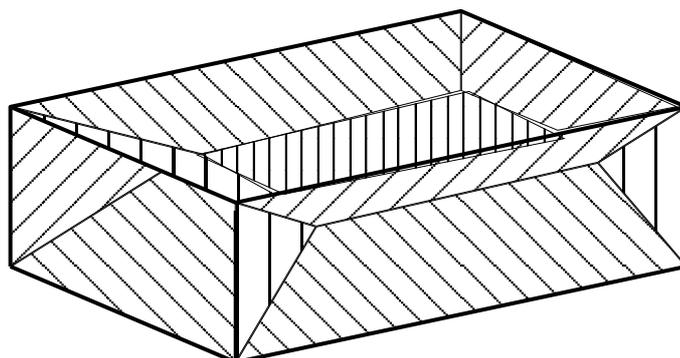


FIG. 1.2 – Squelette d'un parallélépipède.

englobante contient d'autres sommets et dans la négative on poursuit la triangulation en conservant le triangle formé par ces trois sommets.

Une telle 'déviation' de la notion d'axe médian paraît assez arbitraire et ne permet pas de justifier son emploi. C'est pourquoi nous n'avons pas poussé plus avant dans cette direction.

Notons que d'autres définitions de squelettes ont été proposées et étudiées en 2D ([Ros86, Pon88]) mais elles s'appliquent à des objets tellement particuliers (enveloppe de famille à un paramètre de cercles, balayage d'une courbe par un segment...) qu'elles sont inutilisables dans la pratique.

Dans le chapitre suivant nous exposons une autre méthode pour associer une structure filaire à un objet. Il s'agira d'un diagramme, parfois appelé diagramme de Reeb, issu de la théorie de Morse [Gra71] sur les surfaces. Cette notion est d'ailleurs évoquée dans la plupart des articles de L. Kunii et Y. Shinagawa tels que [SKK91, SK91, KST92, IKSU92]... Chaque article est accompagné de l'exemple donné figure 1.3 du diagramme de Reeb de la fonction hauteur sur un tore. Nous présenterons un diagramme similaire fondé sur une autre application que la hauteur. Pour arriver à cette notion nous développons d'abord quelques idées d'ordre général d'une manière analogue à la première partie.

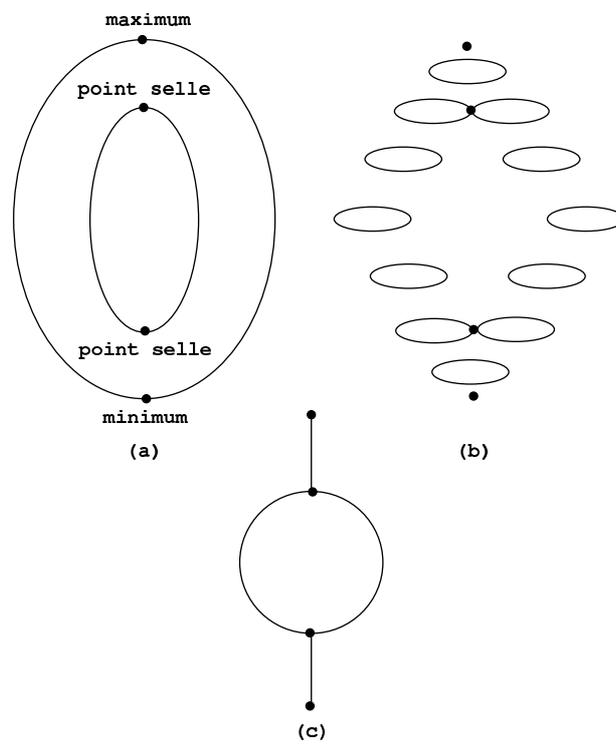


FIG. 1.3 – (a) Un tore, (b) ses sections et (c) son diagramme de Reeb.

Chapitre 2

Vers un diagramme de Reeb

L'objectif de ce chapitre est de justifier sur un plan mathématique l'approche algorithmique du chapitre 3. Nous montrons ici comment relier des concepts de géométrie différentielle tels que les fonctions de Morse ou les espaces fibrés avec une notion d'axe relatif à une surface. Par la suite nous dénoterons X cette surface. Elle représente un objet susceptible d'être métamorphosé.

À plusieurs reprises nous avons évoqué les notions d'espace fibré et de groupe de transformations agissant sur une variété. Ces deux notions fondamentales de la géométrie différentielle peuvent être reliées de la manière suivante. Si nous considérons un sous-groupe G du groupe des difféomorphismes $Dif(X)$ sur une variété X , nous pouvons regarder les orbites de chaque point x de cette variété, c'est-à-dire les ensembles de la forme $G.x$. Dans certains cas l'espace quotient X/G des orbites peut être muni d'une structure de variété différentielle, appelée variété des orbites, telle que X soit un espace fibré au dessus de la base X/G .

L'action d'un groupe permet donc de décomposer X en des éléments plus simples (les fibres et la base). Le cas le plus simple d'une telle action est celui où G est un groupe aux propriétés particulières appelé *groupe proprement discontinu et sans point fixe*. Sans entrer dans les détails (qui sont clairement exposés dans [BG87]) cela signifie que les orbites sont formées de points isolés. Parmi les exemples très classiques d'action de tels groupes sur des ensembles nous avons :

1. le tore T^n obtenu par quotient de \mathbb{R}^n par le groupe des translations engendré par une base de \mathbb{R}^n ,
2. la bouteille de Klein obtenue par quotient de \mathbb{R}^2 par le groupe engendré par les deux translations canoniques et la symétrie $(x, y) \mapsto (x + 1/2, -y)$,
3. les espaces projectifs obtenus par le quotient d'une sphère par le groupe formé de

l'identité et de la symétrie centrale (antipodie).

De tels groupes fournissent des variétés de même dimension que X (qui est alors un revêtement de X/G). Or nous aimerions obtenir une courbe à partir d'une surface et donc pouvoir changer de dimension.

De manière plus générale si G est un groupe de Lie opérant différentiablement sur X , alors sous certaines conditions traduisant grossièrement le fait que les orbites de G sont des sous-variétés de X (c.f. [Die71a] p. 58), X/G peut être muni d'une structure de variété différentielle¹ de dimension $\dim X - \dim G$.

Ainsi la connaissance d'un groupe de Lie de dimension 1 agissant convenablement sur notre objet permet de lui associer une variété de dimension $\dim X - \dim G = 2 - 1 = 1$, c'est à dire une courbe.

Par ailleurs la donnée d'un tel groupe de difféomorphisme à un paramètre (valant l'identité en 0) équivaut à la donnée d'un champ de vecteurs sur la variété X (c.f. [BG87], §3.5). Malheureusement le groupe engendré n'est pas nécessairement convenable. Choisissons par exemple sur \mathbb{R}^2 un champ uniforme du type $\mathbf{e}_1 + \alpha \mathbf{e}_2$ où α est irrationnel. Par passage au quotient sur le tore $T^2 (= \mathbb{R}^2 / \mathbb{Z}^2)$ on obtient un champ dont les courbes intégrales sont denses sur T^2 . Les orbites du groupe à un paramètre correspondant ne sont donc pas des sous-variétés de T^2 .

Un moyen simple de contraindre le champ sur X est de le définir à partir d'une fonction de Morse (i.e. une fonction scalaire dont les singularités sont non dégénérées).

Si

- X est une variété compacte plongée dans \mathbb{R}^3 (munie de la métrique induite),
- J dénote la rotation de $+\pi/2$ dans les espaces tangents (X est orientable) et
- f est une fonction de Morse sur X ,

alors $J \cdot \nabla f$ est un champ sur X dont les courbes intégrales sont les courbes de niveau de f (ou encore f est une intégrale première de $J \cdot \nabla f$). En ôtant de X les images réciproques des valeurs singulières de f (qui sont en nombre fini) on obtient, sur cet ensemble, un groupe G à un paramètre convenable pour le passage au quotient. Ce quotient X/G est donc une courbe (ou plus exactement un ensemble de courbes). Cette courbe est connexe si f admet deux singularités, son minimum et son maximum, et X est alors homéomorphe à une sphère (remarquons que f admet au moins son maximum et son minimum comme valeurs singulières sur une variété compacte).

1. On dit alors que X est un espace fibré principal de groupe structural G .

Les objets sur lesquels sont définis nos métamorphoses sont homéomorphes à la sphère S^2 (une partie cylindrique fermée par deux calottes sphériques). Pour associer une courbe à nos objets nous pouvons donc essayer de définir sur ceux-ci une fonction numérique n'admettant que deux singularités. D'ailleurs, puisque nos objets sont homéomorphes (et même difféomorphes) à la sphère, il existe en théorie un difféomorphisme ψ qui envoie chaque objet sur la sphère, et la fonction recherchée est simplement la composition de ce difféomorphisme avec la fonction hauteur sur la sphère (c.f figure 2.1).

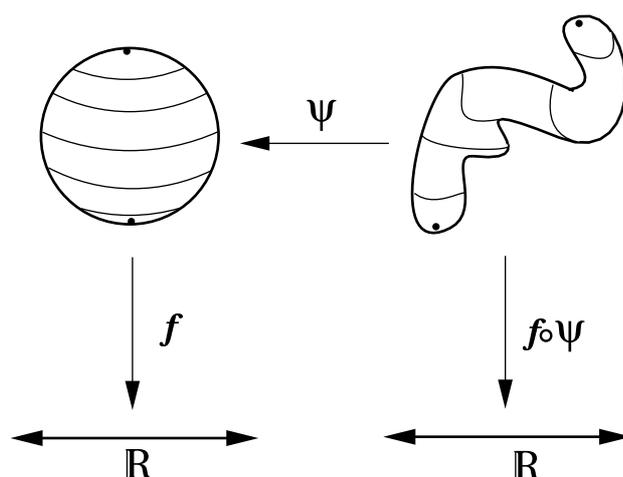


FIG. 2.1 – La fonction de Morse hauteur sur la sphère se transporte sur les formes qui lui sont difféomorphes.

Notons cependant que ce procédé de construction ne nous donne pas tout car la courbe quotient est définie de manière abstraite et elle n'est pas plongée canoniquement dans l'espace \mathbb{R}^3 où évoluent nos objets. Pour cela, il faut trouver un moyen d'associer un point de \mathbb{R}^3 à chaque orbite (ou fibre) de X/G . On peut par exemple prendre le barycentre des orbites ou toute autre application qui associe un point à une courbe fermée (les courbes de niveau de la fonction de Morse). Nous n'avons malheureusement pas trouvé de solution qui fournissait ne serait-ce qu'une immersion en presque tout point de X/G . Dans le cas du barycentre on ne peut espérer mieux comme le montre l'exemple d'une sphère dont on a "enfoncé" un morceau de calotte supérieure (c.f figure 2.2). Par ailleurs la relation d'équivalence sur X induite par les orbites de G permet par passage au quotient sur l'espace topologique sous-jacent à X d'obtenir un espace topologique, également noté X/G (c.f. [Sch70], §VII.4) qui est l'espace sous-jacent au diagramme de Reeb. Les points de ce diagramme s'identifient aux composantes connexes des lignes de niveau de f . De plus ce diagramme possède une structure de complexe simplicial de dimension un.

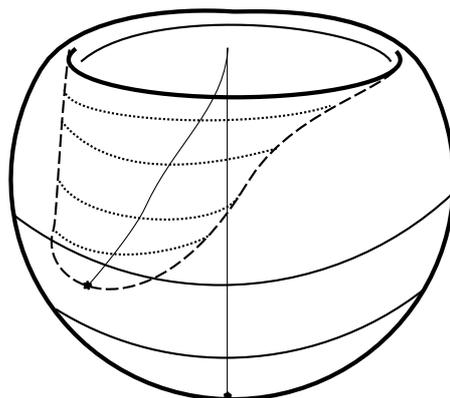


FIG. 2.2 – Les barycentres des lignes de niveau fournissent une courbe qui présente un point de rebroussement à l'endroit où la calotte est enfoncée.

Néanmoins je ne pense pas qu'il existe une immersion canonique de ce graphe dans l'espace ambiant \mathbb{R}^3 . Par exemple les barycentres des composantes connexes des courbes de niveau ne permettent pas de conserver la connexité lorsqu'on passe au voisinage d'un niveau critique (i.e. dont l'image est une valeur critique) comme le montre la figure 2.3. Dans un

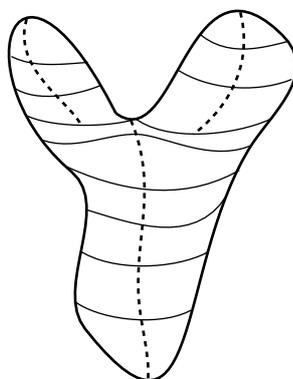


FIG. 2.3 – Les barycentres des composantes connexes des courbes de niveau ne forment pas un ensemble connexe.

premier temps nous nous sommes donc restreints à définir une fonction de Morse sur un objet et à décomposer l'objet en lamelles homéomorphes à un cylindre correspondant aux bouts de surface compris entre deux courbes de niveau.

L'idéal serait de pouvoir construire canoniquement une fonction de Morse n'ayant que deux singularités, ce que nous ne savons pas faire a priori. Nous pouvons par contre essayer de définir une fonction qui soit régulière sur une partie cylindrique aussi étendue que possible

sur l'objet, de sorte que cette fonction fournisse un axe relatif à une grande partie de l'objet.

Comme nous l'avons vu, les fonctions hauteur sont en général des fonctions de Morse. On montre même que ce sont génériquement de telles fonctions [Ker92]. Mais pour qu'une fonction hauteur soit intéressante il faut que l'objet soit *monotone*² par rapport à la direction de projection relative à la hauteur. Or ce n'est pas toujours possible : sur la figure 2.4 nous avons un objet présentant une concavité en forme de demi-sphère qui n'est monotone par rapport à aucune direction. Autrement dit, dès que l'application qui à un point de l'objet

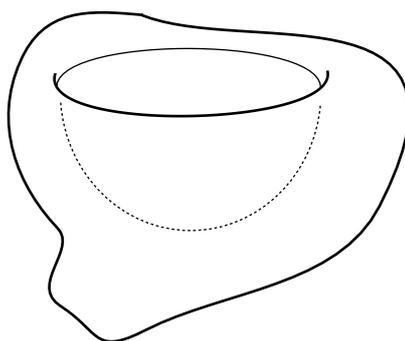


FIG. 2.4 – Un objet qui n'est monotone par rapport à aucune direction.

associe la direction de sa normale, revêt plus de deux fois le plan projectif, on peut dire qu'il n'existe pas de fonction hauteur n'ayant que deux singularités. De plus les fonctions hauteurs dépendent du plongement de l'objet ce qui diminue leur caractère général.

Une fois de plus l'utilisation des distances géodésiques (à un point source donné) semble être appropriée. Dans ce cas les lignes de niveau de l'application distance sont des cercles géodésiques relatifs au point source. En effet pour nos objets qui sont de forme allongée, on peut choisir comme point source un point qui représente intuitivement une extrémité de l'objet. L'étendue de validité de l'application distance est alors au moins égale à l'image du plus grand disque, dans l'espace tangent à la source, où l'application exponentielle est de rang 2. D'une autre manière le rayon de ce disque est la distance géodésique du point source s à son "cut-locus", c'est à dire la distance géodésique minimale aux points conjugués de s ³ (c.f [BG87] §11.4 et [Wal76]). Dans [BG87], §11.4, ce rayon s'appelle *le rayon d'injectivité de l'application exponentielle en s* .

L'hypothèse que nous ferons est donc qu'il existe un point source sur l'objet dont le rayon d'injectivité est grand par rapport au diamètre (géodésique) de

2. sur les courbes et surface monotones en graphique on peut consulter [Paj92].

3. On sait par ailleurs que le cut-locus est l'adhérence des points joints par au moins deux chemins de longueur minimale.

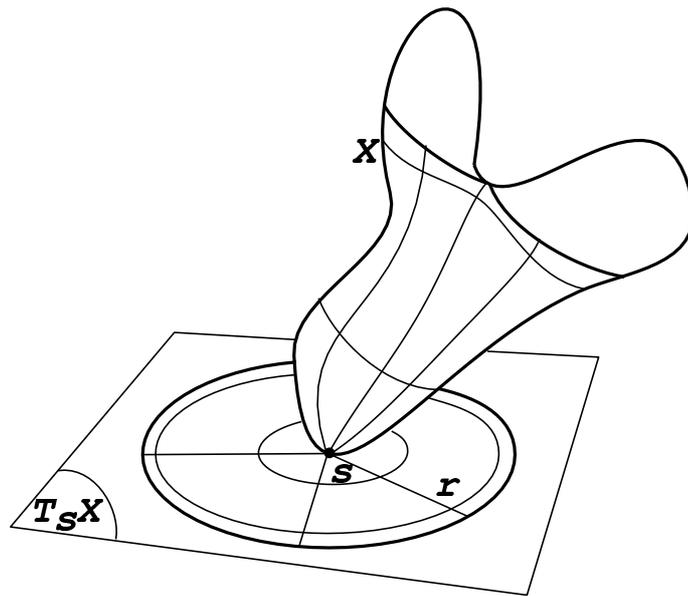


FIG. 2.5 – r est le rayon d'injectivité du point s .

l'objet.

Dans le chapitre suivant nous rappelons les méthodes pour calculer la distance géodésique sur un polyèdre puis nous donnons une description des courbes de niveau pour une fonction numérique définie sur les sommets d'un polyèdre et nous donnons un algorithme de calcul des courbes de niveau relatives à la fonction "distance géodésique à un point source".

Chapitre 3

Un algorithme de découpage en bandelettes

3.1 Introduction

Nous allons, dans ce chapitre¹, décrire des algorithmes pour calculer les courbes de niveau de la distance géodésique à un point source sur un objet polyédrique, afin d'en déduire un axe pour cet objet par moyennage des courbes de niveau. Les deux problèmes principaux à résoudre sont donc :

1. le calcul des distances géodésiques sur un polyèdre,
2. la détermination des lignes de niveau.

Le premier point a largement été traité et nous situerons, dans la section 3.2, notre approche par rapport aux travaux existants. Par contre la détermination des lignes de niveau d'une fonction numérique sur un polyèdre, bien que couramment utilisée dans le domaine de la visualisation scientifique, ne semble pas avoir été explorée de manière précise en liaison avec les aspects topologiques et combinatoires. C'est donc sur ce point que nous insisterons.

Dans la section 3.3.2 nous nous plaçons dans un cadre général où la fonction définie sur l'objet polyédrique est quelconque. Ceci nous permet par analogie avec la théorie de Morse (voir par exemple [Gra71]) de développer une version (très simplifiée) combinatoire de cette théorie. Nous en déduisons chemin faisant une décomposition cylindrique des polyèdres, d'où le titre de ce chapitre.

Puis dans un second temps, dans la section 3.4, nous nous plaçons dans le cas où la fonction est la distance géodésique à un point source sur des polyèdres de topologie sphérique.

1. Une version réduite de ce chapitre a été présentée dans [LV95].

Ceci nous permet dans la section 3.5 de définir un axe pour l'objet polyédrique à partir des courbes de niveau de la distance géodésique à un point source.

3.2 Géodésiques sur les polyèdres

La détermination des géodésiques dans des environnements polyédriques a fait l'objet de nombreuses recherches en particulier pour des applications robotiques. L'application exponentielle (liée aux géodésiques, comme nous l'avons vu dans première partie de ce manuscrit) a également été utilisée dans le but d'interpoler des données sur un polyèdre : possédant des données en certains sommets on étend ces données à tous les sommets à l'aide d'une interpolation liée à un paramétrage local donné par l'application exponentielle ([OP94]).

Le problème fondamental traité dans [MMP87] est de trouver un plus court chemin tracé sur un polyèdre triangulé entre deux sommets quelconques et, plus généralement, pour un sommet source donné, de trouver les plus courts chemins entre ce point source et un point quelconque du polyèdre.

L'idée essentielle intervenant dans la résolution de ce problème est d'utiliser le fait que si γ est un plus court chemin entre deux sommets p et q alors en "dépliant" dans un plan les faces traversées par γ le long des arêtes traversées², γ se déplie en un segment de droite reliant les deux sommets "dépliés" correspondant à p et q (voir figure 3.1).

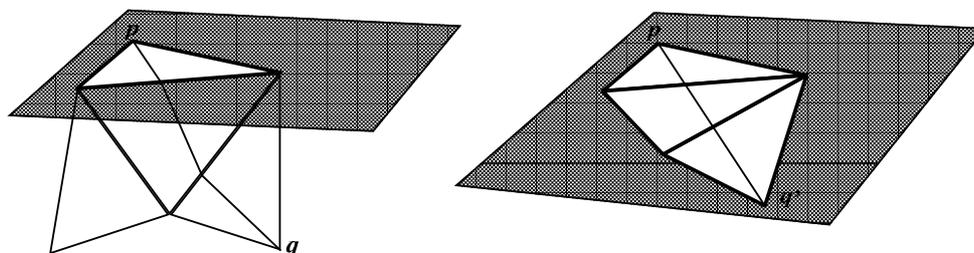


FIG. 3.1 – Les géodésiques se déplient en segment de droite. . .

Dans [MMP87] on remarque ensuite que toute arête peut être décomposée en zones appelées *intervalles d'optimalité* correspondant à des points dont les plus courts chemins (vers la source) associés traversent les mêmes séquences de faces et d'arêtes (voir figure 3.2).

2. Si le polyèdre considéré est convexe, le plus court chemin traverse les arêtes en leur intérieur, sinon il se peut que le plus court chemin passe par des sommets intermédiaires. Dans ce cas la propriété de dépliement rectiligne doit être considérée entre chaque couple de ces sommets.

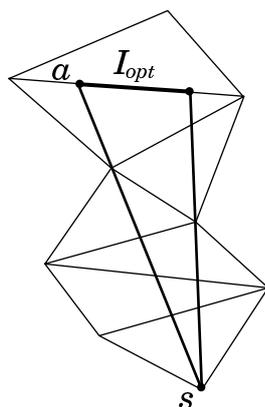


FIG. 3.2 – Propagation d’un intervalle d’optimalité.

L’algorithme donné dans [MMP87] détermine ces intervalles d’optimalité. Comme dans l’algorithme de Dijkstra que nous explicitons plus loin, celui-ci fonctionne sur un principe de propagation des intervalles d’optimalité à partir de la source et maintient une liste d’événements correspondant aux intervalles déjà rencontrés mais qui n’ont pas encore été subdivisés assez finement (i.e. pour lesquels il existe des points dont les plus courts chemins associés passent par des séquences de faces et d’arêtes différentes).

L’algorithme résout le problème des géodésiques avec source unique en temps $\mathcal{O}(n^2 \log n)$ et en place $\mathcal{O}(n^2)$, où n est le nombre d’arêtes du polyèdre. De plus il est possible de trouver les géodésiques à la source pour un point quelconque du polyèdre (i.e. pour un point intérieur à une face) en propageant les intervalles d’optimalité à l’intérieur des faces : chaque arête donne lieu à une subdivision des deux faces adjacentes en $\mathcal{O}(n)$ régions correspondant à ses intervalles d’optimalité. Pour trouver la distance d’un point quelconque on le localise dans une face puis dans les trois subdivisions relatives à chaque arête de la face et on retient le plus court chemin associé.

Après ces pré-calculs, la distance géodésique pour un point quelconque s’obtient en temps linéaire par rapport au nombre de faces traversées par une géodésique pour rejoindre la source.

Les temps de calculs sont cependant trop importants pour un outil interactif. Nous avons préféré nous limiter à une approximation de la distance géodésique en nous restreignant au graphe du polyèdre. Autrement dit, au lieu de considérer le polyèdre avec ses faces, on ne considère que son armature formée de ses arêtes et sommets.

L’algorithme de Dijkstra est tout à fait adapté pour résoudre ce problème. C’est en effet l’algorithme le plus classique pour trouver les plus courts chemins à origine unique dans un

graphe orienté dont les arêtes sont pondérées positivement³. Il est parfaitement décrit dans [CLR94] et nous ne le présenterons pas ici.

Rappelons seulement qu'il peut être programmé à l'aide d'un tas de Fibonacci et qu'il fournit dans ce cas la distance à la source et une géodésique pour tout sommet en temps $\mathcal{O}(n \log n + p)$ où n est le nombre de sommets et p le nombre d'arêtes.

Dans notre implémentation l'utilisateur peut désigner un sommet quelconque du polyèdre avec la souris.

De plus, si le polyèdre n'est pas triangulé, nous complétons le graphe de chaque face par la clique correspondante comme sur la figure 3.3. On augmente ainsi le nombre de

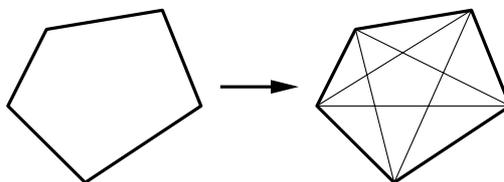


FIG. 3.3 – Les faces sont complétés par toutes arêtes possibles entre leurs sommets.

chemins possibles entre les sommets ce qui permet d'obtenir des distances plus proches de celles obtenues pour des géodésiques tracées sur la *surface* du polyèdre tout en conservant une complexité raisonnable (en admettant que les faces n'ont pas $\mathcal{O}(n)$ arêtes!). Par la suite nous tiendrons compte du fait que les arêtes fournies par l'algorithme de Dijkstra ne sont pas forcément des arêtes du polyèdre mais des liaisons entre deux sommets d'une même face.

3.3 Fonctions numériques sur les polyèdres

Dans toute cette section nous oublions temporairement que la fonction qui sert à définir les courbes de niveau sur nos objets est une distance géodésique.

Nous nous plaçons dans un cadre général où l'objet est un polyèdre triangulé \mathcal{T} et la fonction, notée c , est définie sur les sommets de \mathcal{T} et à valeurs dans un ensemble totalement ordonné quelconque (dans la pratique \mathbb{R}).

Nous supposons que c associe des valeurs distinctes à des sommets adjacents. En théorie des graphes une telle fonction s'appelle un **coloriage** (dans ce cas on ne suppose pas que

3. Il suffit en fait qu'il n'y ait aucun cycle de poids négatif.

l'ensemble d'arrivée est ordonné)⁴. Dans ce qui suit, nous ferons jouer aux coloriage sur les polyèdres un rôle analogue aux fonctions de Morse sur les surfaces⁵.

Il existe en effet une formule simple reliant les indices des points critiques d'une fonction de Morse définie sur une surface et la caractéristique d'Euler-Poincaré de cette surface.

De plus une fonction de Morse permet de découper une surface en morceaux homéomorphes à des cylindres en ôtant de la surface les courbes de niveau (ou plus exactement les ensembles de niveau) relatives à chaque valeur singulière (pour un exposé rapide et clair sur ce sujet on peut se reporter à [Gra71]) comme sur la figure 3.4.

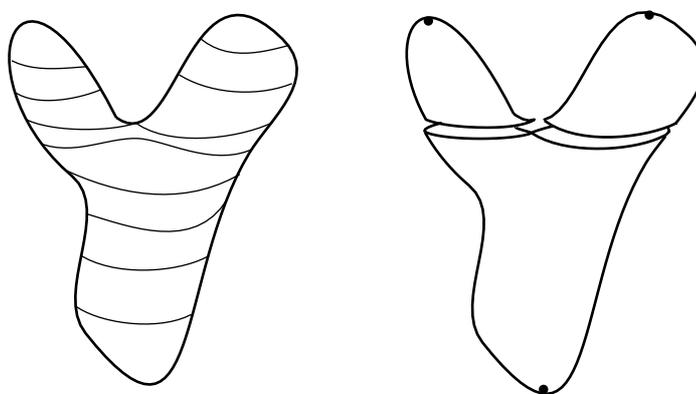


FIG. 3.4 – Un découpage en parties cylindriques. Les points noirs représentent des ensembles critiques qui doivent être ôtés.

Nous montrons dans le reste de la section 3.3 comment ces deux propriétés se traduisent en terme de coloriage sur les polyèdres.

4. Un problème classique lié aux coloriages est par exemple de trouver pour un graphe donné le plus petit cardinal des ensembles image de tous les coloriages possibles de ce graphe [Ber70, AH80].

5. Un cas trivial de coloriage est celui où toutes les valeurs prises par les sommets sont différentes. On en déduit un résultat analogue au fait que les fonctions hauteur sont génériquement des fonctions de Morse. En effet, pour un polyèdre plongé dans \mathbb{R}^3 , l'ensemble des directions de hauteur pour lesquelles les sommets ne reçoivent pas tous des valeurs différentes est l'ensemble des directions orthogonales aux faisceaux de plans déterminés par deux sommets quelconques du polyèdre. Cet ensemble constitue une réunion d'au plus $n(n+1)/2$ sous-variétés compactes de dimension 1 ($= P^1 \sim S^1$), correspondant à chaque faisceau, dans l'ensemble des directions. L'ensemble des directions étant une variété de dimension 2 (le plan projectif), le complémentaire des directions non-admissibles (une union finie de compact d'intérieur vide) est donc bien un ouvert dense dans l'ensemble des directions de hauteur.

3.3.1 Coloriage et caractéristique d'Euler-Poincaré

Nous donnons une relation simple entre le coloriage d'un polyèdre triangulé \mathcal{T} et la caractéristique d'Euler-Poincaré de ce polyèdre. Notons qu'il s'agit d'une relation purement topologique et que le polyèdre n'a pas besoin d'être plongé. Un polyèdre triangulé (topologique) est la donnée d'un graphe non orienté et d'un ensemble de faces qui sont des cycles d'ordre 3 du graphe, tel que l'ensemble des faces adjacentes à un sommet soit homéomorphe à une triangulation du disque de dimension 2. Ce polyèdre n'est pas nécessairement orientable et nous ne précisons cette condition que lorsqu'elle sera nécessaire.

Donnons tout d'abord quelques définitions :

- Si $\bar{c} = (c_1, c_2, \dots, c_n)$ est un n -uplet de nombres, nous appelons **variation** de \bar{c} par rapport au nombre c_0 , et notons $V(\bar{c}|c_0)$ le nombre de changements de signe de la séquence $(sg(c_1, c_0), sg(c_2, c_0), \dots, sg(c_n, c_0))$ où

$$\begin{cases} sg(a, b) = 1 & \text{si } a < b \\ sg(a, b) = -1 & \text{sinon} \end{cases}$$

Exemples :

$$V(1, 2, 3, 4 | 0) = 0$$

$$V(-2, 3.5, 0, -1 | -1.5) = 1$$

- Pour tout sommet s de \mathcal{T} on définit la **variation du sommet s relativement au coloriage c** , notée $V_c(s)$:

si s_1, s_2, \dots, s_n sont les voisins de s ordonnés circulairement (c.f. figure 3.5) alors

$$V_c(s) = V(c(s_1), c(s_2), \dots, c(s_n), c(s_1) | c(s))$$

Si c est à valeurs dans \mathbb{R} c'est aussi le nombre de changements de signe de la séquence $(c(s_1) - c(s), c(s_2) - c(s), \dots, c(s_n) - c(s), c(s_1) - c(s))$ (qu'on appelle souvent aussi variation).

- Nous appelons **indice** du sommet s le nombre (entier) :

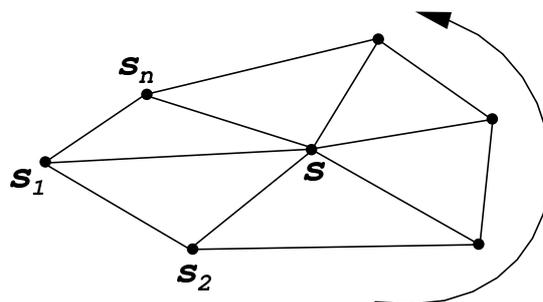
$$Ind_c(s) = 1 - \frac{V_c(s)}{2}$$

Nous avons alors la propriété remarquable :

Proposition 3.1 *Soit \mathcal{T} un polyèdre triangulé, c un coloriage de \mathcal{T} , alors*

$$\sum_{s \in \mathcal{T}} Ind_c(s) = \chi(\mathcal{T}) \tag{3.1}$$

où $\chi(\mathcal{T})$ est la caractéristique d'Euler-Poincaré du polyèdre \mathcal{T} [GS94, HP94].

FIG. 3.5 – Les voisins du sommet s .

Démonstration :⁶ Remarquons tout d’abord qu’un coloriage c induit une orientation sur les arêtes du polyèdre \mathcal{T} (le contraire est faux : c.f. figure 3.6) :

Si a est une arête de \mathcal{T} d’extrémités s_1 et s_2 , a est orientée de s_1 vers s_2 si $c(s_1) < c(s_2)$.

Par ailleurs, en notant respectivement $|s|$, $|a|$ et $|f|$ les nombres de sommets, d’arêtes et de faces de \mathcal{T} , nous avons :

$$\sum_{s \in \mathcal{T}} \text{Ind}_c(s) = |s| - \frac{1}{2} \sum_{s \in \mathcal{T}} V_c(s)$$

Or $\sum_{s \in \mathcal{T}} V_c(s)$ compte le nombre de paires d’arêtes incidentes à une même face qui sont orientées de manière compatible. Comme chaque face possède exactement une paire d’arêtes de ce type (c.f. figure 3.7), ce nombre est également le nombre de faces de \mathcal{T} , d’où :

$$\sum_{s \in \mathcal{T}} \text{Ind}_c(s) = |s| - \frac{1}{2}|f|$$

Par ailleurs, \mathcal{T} étant triangulé, nous avons

$$3|f| = 2|a|$$

qui est encore le nombre de paires (f, a) où l’arête a est incidente à la face f . On en déduit

$$\sum_{s \in \mathcal{T}} \text{Ind}_c(s) = |s| - |a| + |f| = \chi(\mathcal{T})$$

qui est la classique formule d’Euler \square

Pour justifier le terme d’*indice* nous proposons de nous ramener à la notion d’indice d’un champ de vecteurs. Nous définissons à cette fin un champ de vecteurs continu sur \mathcal{T} tangent à ses arêtes (orientées) et dont les singularités (i.e. là où le champ est nul) sont exactement les sommets de \mathcal{T} . Pour cela il suffit de le définir sur une face (triangulaire) de \mathcal{T} de sorte que les triangles se “raccordent” continûment.

6. Je remercie Christian Choffrut et Pascal Lienhardt de m’avoir communiqué cette démonstration.

Soit $f = (s_1, s_2, s_3)$ une face de \mathcal{T} telle que $c(s_1) < c(s_2) < c(s_3)$. Il est toujours possible de ré-indexer les sommets de f pour que cette relation soit vraie. On définit un champ sur f dont les lignes de champs ont l'allure de la figure 3.6. Nous donnons en appendice F un exemple de construction de ce champ (qui n'est pas si simple qu'elle paraît). Le champ

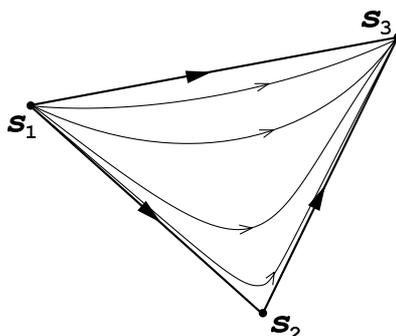


FIG. 3.6 – Les lignes de champ sur un triangle. La configuration représentée a toujours lieu : une arête a une orientation opposée aux deux autres, plus généralement il n'existe aucun cycle d'arêtes d'orientations compatibles sur le polyèdre.

au voisinage d'un sommet s de \mathcal{T} a par exemple l'allure de la figure 3.7. Il est facile de

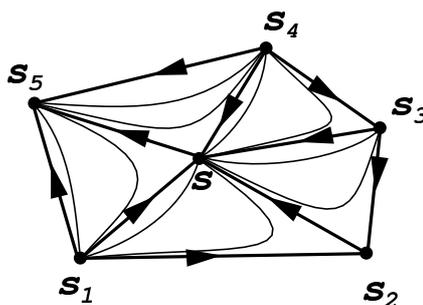


FIG. 3.7 – L'indice du sommet s est nul.

voir que l'indice de ce champ au sommet s est justement ce que nous avons appelé l'indice d'un sommet. Nous retrouvons ainsi la relation entre les indices des singularités d'un champ de vecteurs et la caractéristique d'Euler-Poincaré d'une surface (c.f. [Pha92] : Indices des champs de vecteurs et caractéristique d'Euler des surfaces, pp. 85-98).

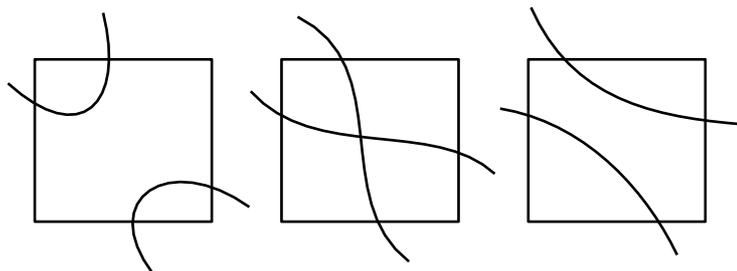


FIG. 3.9 – Les trois possibilités de raccord des lignes de niveau dans un rectangle dégénéré.

Lorsqu'il n'y a pas d'informations particulières sur la fonction à interpoler, le point s d'intersection de la ligne de niveau h avec une arête (s_1, s_2) est obtenu par interpolation linéaire inverse :

$$s = \alpha s_1 + (1 - \alpha) s_2, \quad \alpha = \frac{h(s_2) - h}{h(s_1) - h(s_2)}$$

Puis, le contour est interpolé entre ces sommets s , soit linéairement soit à l'aide d'une spline (c.f. figure 3.10).

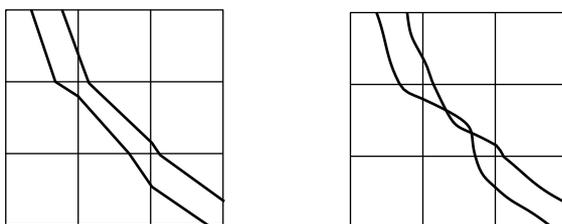


FIG. 3.10 – À gauche : interpolation linéaire des courbes de niveau. À droite : l'interpolation spline peut générer des incohérences.

L'interpolation par une spline s'étend plus difficilement à des contours tracés sur des polyèdres car la spline d'interpolation n'a pas de raison de coller au polyèdre. La solution proposée dans [Wij91] est de calculer directement un "rendu" de la surface polyédrique avec ses courbes de niveau. Pour cela l'auteur associe une couleur à chaque point intérieur aux faces du polyèdre en fonction de sa distance aux sommets de la face le contenant et de la valeur de ces sommets⁷. Un paramètre permet de régler l'épaisseur des courbes de niveau (c.f. [Wij91] pour de jolies photographies couleur).

Cette dernière méthode qui a le mérite d'être robuste, en ce sens qu'elle ne présente

7. Pour un état de l'art sur les interpolations à partir de données discrètes voir [FN94]

pas de difficulté numérique ou algorithmique particulière, ne permet pas d'extraire de façon simple des informations telles que le barycentre sur les courbes de niveau.

Par contre le suivi de contour sur une grille comme dans [Sut87] fournit des lignes polygonales que l'on peut aisément manipuler. C'est pourquoi nous avons choisi ce type d'algorithme et de représentation pour les courbes de niveau des coloriage. Cependant la robustesse des algorithmes de suivi n'est pas traitée et requiert des choix arbitraires comme le montre la figure 3.9.

Dans ce qui suit nous donnons une description plus précise des courbes de niveau pour le coloriage d'un polyèdre connexe, orientable et triangulé \mathcal{T} que nous étendons à des polyèdres non triangulés.

3.3.3 Courbes de niveau et coloriage sur les triangulations

Nous décrivons ici une notion de courbe de niveau sur un polyèdre triangulé et colorié. Tout ce qui suit est, à notre connaissance, entièrement original et ne fait référence à aucun article particulier.

Comme en 3.3 nous nous donnons un polyèdre triangulé \mathcal{T} et un coloriage c à valeur dans \mathbb{R} . Les arêtes de \mathcal{T} sont orientées comme précédemment.

Nous dirons qu'une valeur est **régulière** si elle n'appartient pas à l'image de c . Nous notons M le maximum (resp. m le minimum) de c :

$$M = \max_{s \in \mathcal{T}} c(s)$$

$$m = \min_{s \in \mathcal{T}} c(s)$$

Nous dirons qu'une arête (s_1, s_2) est **transverse** au niveau c_0 si

$$c(s_1) < c_0 < c(s_2)$$

Remarquons par ailleurs que \mathcal{T} induit un graphe sur ses arêtes pour la relation qui lie deux arêtes adjacentes et incidentes à une même face (dans le cas d'une triangulation l'adjacence est automatiquement vérifiée). Ce graphe peut canoniquement être muni d'une structure de polyèdre, les faces correspondant aux cycles d'arêtes incidentes à une même face ou à un même sommet. Il s'agit donc du polyèdre tronqué de \mathcal{T} qui a les mêmes caractéristiques topologiques que ce dernier. Nous notons \mathcal{A} ce polyèdre.

Nous appelons **contour de niveau** c_0 , et notons \mathcal{C}_{c_0} l'ensemble des arêtes (de \mathcal{T}) transverses au niveau c_0 . Ces arêtes forment un sous-ensemble des sommets de \mathcal{A} auquel nous

associons une structure de graphe induite (i.e. \mathcal{C}_{c_0} est considéré comme un sous graphe de \mathcal{A}).

Proposition 3.2 *Pour toute valeur régulière c_0 comprise entre m et M le contour de niveau c_0 est une union finie non vide de cycles simples de \mathcal{A} .*

Démonstration :

1. Il existe au moins une arête transverse au niveau c_0 : soit s_m (resp. s_M) un sommet de couleur m (resp. M). Puisque \mathcal{T} est connexe il existe un chemin (i.e. une suite de sommets adjacents) $(s_m = s_1, s_2, \dots, s_q = s_M)$ joignant s_m à s_M . Posons :

$$j = \max(k \geq 1 \mid c(s_i) < c_0, \forall i = 1, \dots, k)$$

alors l'arête (s_j, s_{j+1}) est transverse à c_0 .

2. Toute arête $a = (s_1, s_2)$ transverse à c_0 est adjacente dans \mathcal{A} à exactement deux arêtes transverses. En effet, chacune des deux faces f_1 et f_2 incidentes à a admet une seule autre arête a_1 (resp. a_2) transverse à c_0 . Il suffit de considérer les trois cas possibles suivants (voir figure 3.11 gauche pour les notations) :

$$\begin{aligned} c(s_3) < c(s_1) < c(s_2) &\Rightarrow a_s = (s_3, s_1) \\ c(s_1) < c(s_3) < c(s_2) &\Rightarrow \text{ou bien } a_s = (s_1, s_3) \text{ ou bien } a_s = (s_3, s_2) \\ c(s_1) < c(s_2) < c(s_3) &\Rightarrow a_s = (s_1, s_3) \end{aligned}$$

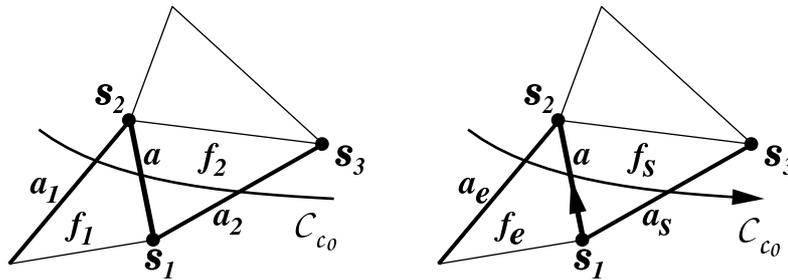


FIG. 3.11 – À gauche : la configuration générale. À droite : le cas orienté.

Comme il y a un nombre fini d'arêtes, \mathcal{C}_{c_0} est nécessairement une union finie de cycles simples \square

Remarque 3.1 Lorsque \mathcal{T} est orientable, son orientation induit une orientation de \mathcal{C}_{c_0} . En effet l'orientation de a et de \mathcal{T} permet de singulariser une face entrante f_e et une face sortante f_s pour a (voir figure 3.11 droite). Chacune de ces faces permet de définir une arête de sortie a_s pour laquelle f_s est une face entrante et une arête entrante a_e pour laquelle f_e est sortante. Chaque arête a transverse possède donc un unique prédécesseur et un unique successeur transverse relativement à l'orientation locale.

Nous numérotons désormais les sommets de \mathcal{T} de 1 à n de sorte que :

$$m = c(s_1) \leq c(s_2) \leq \dots \leq c(s_n) = M$$

Lemme 3.1 Si c_1 et c_2 sont deux niveaux appartenant à un même intervalle $]c(s_{i-1}), c(s_i)[$ ($i \in [2, n]$) alors $\mathcal{C}_{c_1} = \mathcal{C}_{c_2}$. On note \mathcal{C}_i cette valeur commune.

Démonstration : Toute arête transverse à c_1 est transverse à c_2 et réciproquement \square

Si $c(s_{i-1}) < c(s_i) = c(s_{i+1}) = \dots = c(s_{i+p}) < c(s_{i+p+1})$ on posera :

$$\mathcal{C}_{i+p} = \mathcal{C}_{i+p-1} = \dots = \mathcal{C}_i$$

On en déduit que les changements de topologie des lignes de niveau ne peuvent se produire qu'au passage des sommets. Ces changements dépendent clairement de l'indice des sommets considérés (c.f. figure 3.12).

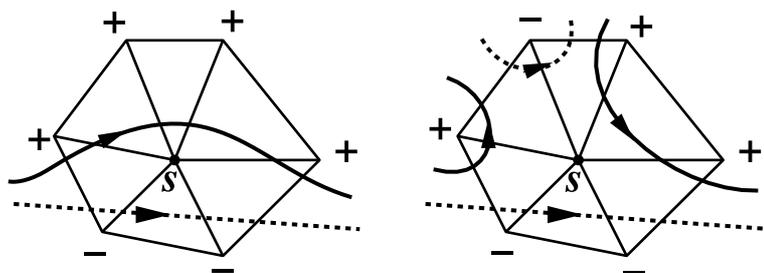


FIG. 3.12 – Deux exemples de modifications de contour au voisinage d'un sommet s . Les “+” (resp. “-”) symbolisent des sommets de couleurs supérieures (resp. inférieures) à $c(s)$. La courbe en trait pointillé (resp. plein) représente la ligne de niveau inférieure (resp. supérieure) à $c(s)$.

Proposition 3.3 Si tout sommet de même couleur que s_i a un indice nul alors \mathcal{C}_i et \mathcal{C}_{i+1} sont homotopes. Ils sont alors constitués du même nombre de cycles qui se correspondent deux à deux et se déduisent par substitution de chaînes simples.

Démonstration : Soit $a = (\sigma_1, \sigma_2)$ une arête orientée de \mathcal{C}_i .

- ou bien $c(\sigma_2) \neq c(s_i)$ et alors $a \in \mathcal{C}_{i+1}$,
- ou bien $c(\sigma_2) = c(s_i)$ et alors

$$\text{Ind}_c(\sigma_2) = 0 \quad \text{c.a.d} \quad V_c(\sigma_2) = 2$$

Les voisins de σ_2 forment (dans \mathcal{T}) un cycle simple ayant deux composantes connexes : les voisins de couleur supérieure à $c(s_i)$ et ceux de couleur inférieure (voir figure 3.13). Les arêtes reliant σ_2 à ses voisins “inférieurs” forment donc une chaîne simple dans

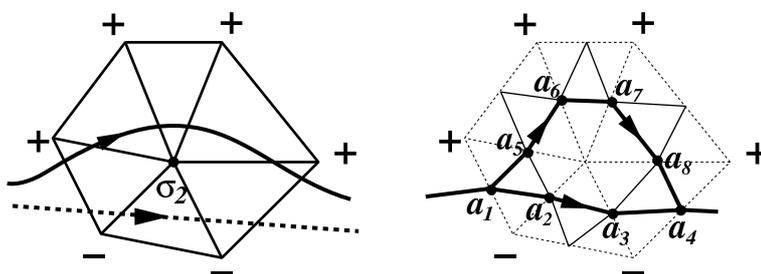


FIG. 3.13 – À droite : le sommet σ_2 de couleur $c(s_i)$ d'indice nul. À gauche : le graphe des arêtes correspondantes. \mathcal{C}_i contient la chaîne (a_1, a_2, a_3, a_4) qui est remplacée par $(a_1, a_5, a_6, a_7, a_8, a_4)$ dans \mathcal{C}_{i+1} .

\mathcal{C}_i qui contient a . Cette chaîne est remplacée dans \mathcal{C}_{i+1} par les arêtes reliant σ_2 à ses voisins “supérieurs” \square

On appellera **sommets réguliers** les sommets d'indice nul et **sommets critiques** les autres sommets.

Remarque 3.2 Les sommets d'indice positif (nécessairement égal à 1) sont des *extremums locaux*. La formule d'Euler de la section précédente devient ainsi :

$$\#\{\text{extremum local}\} + \sum_{\text{Ind}(s) < 0} \text{Ind}(s) = \chi(\mathcal{T})$$

De plus, l'évolution des contours de niveau au voisinage d'un extremum se traduit localement par des créations ou disparitions de cycles.

On appelle **réalisation** d'un contour $\mathcal{C}_h = (a_1, a_2, \dots, a_q)$ ou plus simplement **ligne de niveau**, notée \mathcal{L}_h , une ligne polygonale (p_1, p_2, \dots, p_q) telle que p_i appartient à a_i et dont les cycles correspondent aux cycles de \mathcal{C}_h .

Exemples de réalisation

1. Si c est une application hauteur, les lignes de niveau sont les intersections de \mathcal{T} avec les plans perpendiculaires à la direction de hauteur. Le sommet p de \mathcal{L}_h sur une arête $a = (s_1, s_2)$ de \mathcal{C}_h est obtenu par la formule :

$$p = \alpha s_1 + (1 - \alpha) s_2, \quad \alpha = \frac{c(s_2) - h}{c(s_1) - c(s_2)} \quad (3.2)$$

2. Si c est une application distance déduite de l'algorithme de Dijkstra, nous conservons la formule 3.2. h coïncide d'ailleurs avec la distance géodésique du point p de l'arête (s_1, s_2) à la source (relativement à l'armature du polyèdre \mathcal{T}), si et seulement si il existe un chemin géodésique pour s_2 contenant l'arête (s_1, s_2) .

En effet, soit γ_1 un chemin géodésique joignant s_1 à la source, en notant c_p la distance géodésique de p à la source, nous avons :

$$c_p = c(s_1) + d(s_1, p) = c(s_2) - d(p, s_2)$$

puisque $s_2 s_1 \cdot \gamma_1$ est une géodésique, ou encore :

$$c(s_2) = c(s_1) + d(s_1, s_2) = c(s_1) + d(s_1, p) + d(p, s_2)$$

Comme par ailleurs

$$p = s_1 + \frac{d(s_1, p)}{d(s_1, s_2)} (s_2 - s_1).$$

on en déduit la formule 3.2 avec

$$\alpha = \frac{c(s_2) - c_p}{c(s_1) - c(s_2)}$$

c'est à dire $h = c_p$.

Si $s_2 s_1 \cdot \gamma_1$ n'est pas une géodésique alors $c(s_2) < c(s_1) + d(s_1, s_2)$ et la formule précédente est fausse.

3.3.4 Décomposition en bandelettes

On déduit de ce qui précède la décomposition suivante :

Proposition 3.4 *Soit un polyèdre connexe et triangulé \mathcal{T} , c un coloriage de \mathcal{T} et s_1, s_2, \dots, s_n les sommets de \mathcal{T} ordonnés par leur couleur, une décomposition de \mathcal{T} en bandelettes homéomorphes à des (unions de) cylindres est donnée par l'union des bandes B_i où*

$$B_i = \bigcup_{c(s_i) < h < c(s_{i+1})} \mathcal{L}_h$$

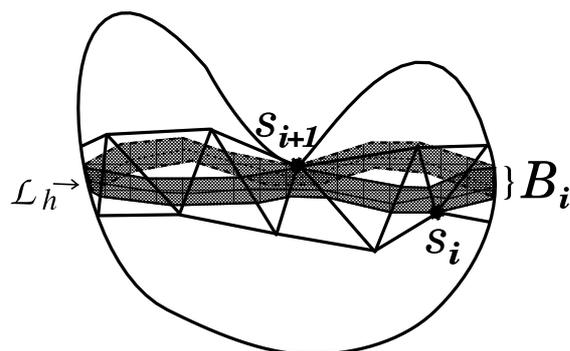


FIG. 3.14 – Une bande cylindrique entre deux niveaux $c(s_i)$ et $c(s_{i+1})$.

La figure 3.14 montre une bande B_i formée d'un seul cylindre. Il est possible d'obtenir une décomposition minimale en regroupant les cylindres de la décomposition précédente. Chaque B_i est en effet composée de plusieurs cylindres, autant que de composantes connexes de \mathcal{C}_i , que nous notons $\mathcal{D}_{i,j}$. On regroupe alors $\mathcal{D}_{i,j}$ et $\mathcal{D}_{i+1,k}$ s'ils ont un bord commun et si les éventuels sommets (de \mathcal{T}) de ce bord sont réguliers.

3.3.5 Cas des polyèdres non triangulés

Lorsque \mathcal{T} n'est pas triangulé la formule d'Euler 3.1 n'est plus valable mais il est possible de donner une décomposition semblable à la précédente si \mathcal{T} est orientable. Il suffit pour cela d'adapter la notion de contour de niveau :

Si $a = (s_1, s_2)$ est une arête transverse au niveau c_0 nous définissons comme dans la remarque 3.1 une face de sortie f_s et une face d'entrée f_e (voir figure 3.15). L'arête de sortie (resp.

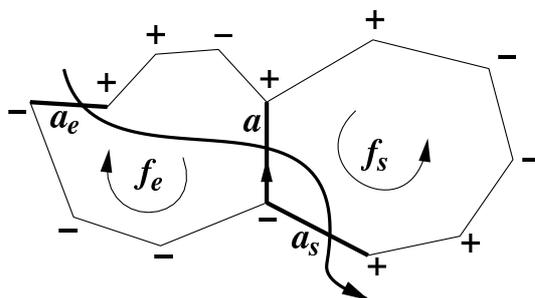


FIG. 3.15 – L'arête de sortie (resp. d'entrée) a_s (resp. a_e) est alors la première arête rencontrée transverse à $c_0 \dots$

d'entrée) a_s (resp. a_e) est alors la première arête rencontrée, transverse à c_0 , en parcourant

les arêtes de f_s (resp. f_e) dans le sens positif (resp. négatif) relativement à l'orientation de a et \mathcal{T} .

Avec cette convention les arêtes a_e , a et a_s ne sont plus adjacentes dans \mathcal{A} mais le sont pour la relation qui lie deux arêtes incidentes à une même face. Ce nouveau graphe est celui de \mathcal{A} complété par les cliques des faces de \mathcal{A} correspondant à des faces de \mathcal{T} . Il n'est a priori pas plongeable dans un polyèdre. Dans ce nouveau graphe, les contours de niveau ne sont plus des cycles simples mais la convention ci-dessus permet de les parcourir une seule fois !

3.3.6 Cas des triangulations non coloriées

Lorsque \mathcal{T} est triangulé mais que la fonction numérique c n'est pas un coloriage (i.e. deux sommets adjacents peuvent avoir la même valeur), les propositions 3.2 et 3.1 sont toujours vraies, mais nous n'avons plus de décomposition en bandelettes (dont l'adhérence recouvre \mathcal{T}). Une fonction constante n'a, par exemple, aucune ligne de niveau puisqu'il n'existe pas de valeur régulière comprise entre les extremums de cette fonction...

Par contre dans le cas particulier de la distance à un point source sur \mathcal{T} , un sommet ne peut avoir tous ses voisins à la même distance. On en déduit une décomposition formée de bandelettes cylindriques et de faces entourées de sommets équidistants de la source, et l'adhérence de cette décomposition recouvre entièrement \mathcal{T} .

3.4 Application au coloriage par la distance géodésique

Nous supposons désormais que la valeur d'un sommet est obtenue par l'algorithme de Dijkstra comme distance géodésique à un sommet source s . Notons que cette spécialisation n'est pas purement anecdotique car l'application distance géodésique à un point source est définie de manière canonique sur une variété riemannienne (complète) pointée. Dans le cadre combinatoire ceci s'interprète par le fait que sur un graphe non plongé ayant des arêtes valuées (l'équivalent de la métrique) et pour lequel on a distingué un sommet, on peut définir algorithmiquement (par l'algorithme de Dijkstra) une distance géodésique.

Nous déduisons des propriétés de l'algorithme de Dijkstra un moyen simple de calculer les contours de niveau.

Remarquons que la distance géodésique d n'est pas nécessairement un coloriage car des sommets adjacents peuvent être à une même distance de la source, mais on peut s'y ramener en utilisant le lemme suivant :

Lemme 3.2 *Pour toute fonction numérique d sur \mathcal{T} on peut en déduire en un temps $\mathcal{O}(n \log n)$*

un coloriage c de sorte que :

$$d(s_i) < d(s_j) \quad \Rightarrow \quad c(s_i) < c(s_j) \quad (3.3)$$

Démonstration : Il suffit d'ordonner les n sommets de \mathcal{T} par ordre de distance croissante. Puis par récurrence on pose :

$$\begin{aligned} c(s_i) &= d(s_i) & \text{si } d(s_i) < d(s_{i+1}) \\ c(s_i) &= d(s_i) - \epsilon & \text{si } d(s_i) = d(s_{i+1}) \end{aligned}$$

avec $\epsilon < d(s_i) - d(s_{i-1})$ \square

Si d est obtenue par l'algorithme de Dijkstra on peut d'ailleurs construire le coloriage lors du calcul des distances puisque dans cet algorithme les sommets sont "découverts" par ordre croissant de distance.

Remarque 3.3 *Un chemin strictement monotone pour d l'est donc pour c . En particulier les géodésiques fournies par l'algorithme de Dijkstra donnent des chemins strictement décroissants pour c , en supposant que deux sommets adjacents de \mathcal{T} n'ont pas les mêmes coordonnées dans \mathbb{R}^3 .*

Dans la pratique nous ne nous intéresserons, pour les métamorphoses, qu'à des objets de topologie sphérique et **nous supposerons désormais que \mathcal{T} a une topologie sphérique.**

Nous avons alors :

Proposition 3.5 *Si le contour de niveau c_0 a p composantes connexes, i.e est constitué de p cycles, alors toute réalisation de \mathcal{C}_{c_0} sépare \mathcal{T} en $p + 1$ composantes connexes. Elles sont formées d'une composante qui contient tous les sommets de couleur inférieure à c_0 , dont la source, et de p calottes bordées par chacune des composantes du contour. De plus ces p calottes contiennent chacune au moins un maximum local.*

Démonstration : Notons $\mathcal{C}^1, \dots, \mathcal{C}^p$ les p composantes de \mathcal{C}_{c_0} et \mathcal{C}_-^0 le graphe restreint aux sommets de couleur inférieure à c_0 . Puisque \mathcal{T} a une topologie sphérique, \mathcal{C}^i sépare \mathcal{T} en deux composantes connexes homéomorphes à des disques. Nous notons \mathcal{C}_+^i celle qui ne contient pas la source s (voir figure 3.16).

- Ces composantes sont disjointes deux à deux. En effet, supposons par l'absurde que :

$$p \in \mathcal{C}_+^i \cap \mathcal{C}_+^j$$

alors l'algorithme de Dijkstra fournit un chemin strictement décroissant de p vers la source qui ne rencontre \mathcal{C}_{c_0} qu'une seule fois. On en déduit que ce chemin ne rencontre pas l'une des deux composantes \mathcal{C}^i ou \mathcal{C}^j ce qui contredit la propriété de séparation de \mathcal{C}^i ou \mathcal{C}^j .

- Chaque \mathcal{C}_+^i contient un maximum local : en effet soit s_i un sommet de \mathcal{C}_+^i qui réalise le maximum de c sur \mathcal{C}_+^i . Un voisin s_j de s_i est ou bien dans \mathcal{C}_+^i , et donc de couleur inférieure, ou bien tel que l'arête (s_j, s_i) est transverse à c_0 et on a nécessairement (voir figure 3.16) :

$$c(s_j) < c_0 < c(s_i)$$

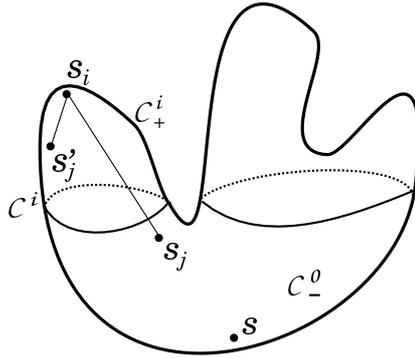


FIG. 3.16 – Les voisins de s_i ont une couleur inférieure à $c(s_i)$.

- \mathcal{C}_-^0 ne rencontre aucun des \mathcal{C}_+^i : chaque sommet de \mathcal{C}_-^0 est en effet joint à la source par un chemin strictement décroissant qui ne rencontre pas les contours \mathcal{C}^i , ce qui montre d'ailleurs que \mathcal{C}_-^0 est connexe.
- Un sommet qui n'est pas dans \mathcal{C}_-^0 , donc de couleur supérieure à c_0 , est dans l'une des \mathcal{C}_+^i : il existe un chemin strictement décroissant (une géodésique) vers la source qui rencontre une fois \mathcal{C}_{c_0} donc l'un des \mathcal{C}^i ; ce sommet est dans la composante déterminée par \mathcal{C}^i qui ne contient pas la source, i.e. dans \mathcal{C}_+^i .

On déduit de ceci que $\mathcal{C}_-^0, \mathcal{C}_+^1, \dots, \mathcal{C}_+^p$ réalise la partition de \mathcal{T} recherchée \square

Lemme 3.3 *Pour tout sommet σ de \mathcal{T} l'ensemble des arêtes incidentes à σ et transverses à un niveau c_0 inférieur à $c(\sigma)$ appartiennent à un même cycle de \mathcal{C}_{c_0} .*

Démonstration : Soient $a_i = (s_i, \sigma)$ et $a_j = (s_j, \sigma)$ deux arêtes transverses au niveau c_0 . Soit \mathcal{C}^i (resp. \mathcal{C}^j) la composante de \mathcal{C}_{c_0} contenant a_i (resp. a_j).

Toute réalisation de \mathcal{C}^i sépare \mathcal{T} en deux composantes connexes dont l'une contient la source s et l'autre σ . Il en est de même pour \mathcal{C}^j . Nous avons donc deux cycles (confondus ou

disjoints) sur une sphère dont les deux composantes respectives ont un point en commun. On en déduit que l'un des cycles, par exemple \mathcal{C}^j , entoure l'autre qui entoure σ (c.f. figure 3.17). L'arête a_j rencontre donc \mathcal{C}^i et par conséquent $\mathcal{C}^i = \mathcal{C}^j$ \square

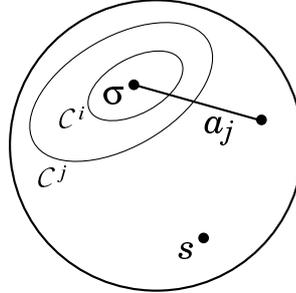


FIG. 3.17 – \mathcal{C}^j entoure \mathcal{C}^i qui entoure σ et \mathcal{C}^i entoure \mathcal{C}^j qui entoure s !

En utilisant les notations de la proposition 3.3 et en notant n_i le nombre de cycles de \mathcal{C}_i , on déduit du lemme précédent proposition suivante :

Proposition 3.6 *Pour tout i tel que $c(s_i) < c(s_{i+1})$ nous avons :*

$$n_{i+1} = n_i - \sum_{c(s_j)=c(s_i)} \text{Ind}_c(s_j)$$

Démonstration : Soient s_{i-p}, \dots, s_i les sommets de couleur $c(s_i)$, on peut par le procédé du lemme 3.2 trouver un coloriage c' tel que $c'(s_{i-p}) < c'(s_{i-p+1}) < \dots < c'(s_i)$ et tel que $n_i = n'_{i-p}$ et $n_{i+1} = n'_{i+1}$. Nous pouvons donc supposer que $c(s_{i-1}) < c(s_i) < c(s_{i+1})$.

- Si $\text{Ind}_c(s_i) = 0$ alors la proposition 3.3 permet de conclure,
- si $\text{Ind}_c(s_i) = 1$ alors la remarque 3.2, section 3.3.3 permet de conclure,
- si $\text{Ind}_c(s_i) < 0$ le lemme 3.3 précédent montre que l'on a nécessairement la configuration de la figure 3.18 (le lecteur pourra généraliser par lui-même...) \square

3.4.1 Algorithme

Nous présentons un algorithme qui, à partir de la donnée d'un polyèdre \mathcal{T} triangulé et orientable, d'un point source et des distances géodésiques associées, calcule progressivement tous les contours \mathcal{C}_i et une représentation associée.

Ceci est effectué en examinant successivement les sommets du polyèdre triés par ordre croissant de distance et en initialisant le contour \mathcal{C}_i avec le cycle des arêtes incidentes à la

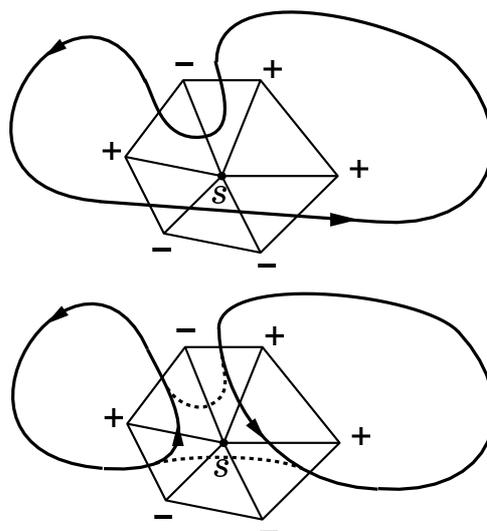


FIG. 3.18 – Au dessus : la ligne de niveau légèrement inférieure à $c(s_i)$. En dessous : en trait plein, la ligne de niveau légèrement supérieure à $c(s_i)$. L'indice du sommet vaut -1 et il y a 1 création de cycle.

source. À chaque nouveau sommet examiné, on réactualise la liste des cycles en fonction de son indice. À la fin de l'algorithme la liste est formée des cycles entourant les maximums globaux.

Par mesure de concision, nous dirons qu'une arête, incidente à un sommet s , est **négative** (relativement à s) si son autre sommet a une couleur inférieure à $c(s)$ et nous dirons qu'elle est **positive** dans le cas contraire. L'ensemble des arêtes incidentes à s forment un cycle ordonné circulairement. Dans ce cycle nous appellerons **chaîne d'arêtes positives** un ensemble maximal d'arêtes positives connectées dans \mathcal{A} .

De plus, en reprenant les notations de la proposition 3.3, pour un sommet s_i , le lemme 3.3 montre que dans le contour \mathcal{C}_i toutes les arêtes négatives de s_i appartiennent à un même cycle. Nous notons \mathcal{C}^i ce cycle.

Les n sommets de \mathcal{T} sont indexés par ordre de distance. L'algorithme se présente comme suit :

Initialisation : le contour courant est initialisé au cycle des arêtes incidentes à la source s_1 .

Boucle principale : pour chaque sommet s_i on détermine le cycle ordonné circulairement de ses arêtes incidentes, on en déduit son indice, puis

1. on met à jour le contour courant en fonction de cet indice :

- si $Ind_c(s_i) = 1$: alors s_i est un maximum local et il faut détruire le cycle des arêtes incidentes à s_i (ce cycle fait nécessairement partie de la liste des cycles du contour courant et coïncide avec \mathcal{C}^i).
- si $Ind_c(s_i) = 0$: alors s_i est un sommet régulier et il faut juste réactualiser le cycle \mathcal{C}^i . Pour cela on substitue la chaîne des arêtes positives de s_i à la chaîne de ses arêtes négatives dans le cycle \mathcal{C}^i (c.f. figure 3.13).
- si $Ind_c(s_i) < 0$: alors s_i est un sommet singulier de type selle. D'après la proposition 3.6 il y a création de $-Ind_c(s_i)$ cycles. Il faut également réactualiser le cycle \mathcal{C}^i . Pour cela, on parcourt la liste des arêtes incidentes à s_i , puis :
 - on substitue la première chaîne d'arêtes positives rencontrée à la chaîne comprise entre les deux arêtes négatives bordantes (c.f. figure 3.19) afin de réactualiser le cycle \mathcal{C}^i ,

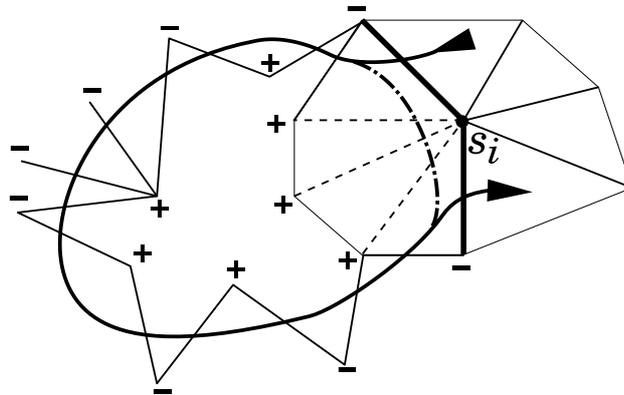


FIG. 3.19 – Les arêtes en pointillés se substituent aux arêtes en trait plein et gras et le cycle est coupé à cet endroit.

- puis, pour chaque nouvelle chaîne d'arêtes positives rencontrée, il faut créer un nouveau cycle avec le reste des arêtes du cycle initial \mathcal{C}^i et faire la substitution précédente.

Par ailleurs pour retrouver le cycle \mathcal{C}^i , dans la liste du contour courant, il est nécessaire de le lier à l'une (et une seule) des arêtes qui le composent. La mise à jour de ce lien dépend de l'indice de s_i :

- si $Ind_c(s_i) = 1$, il suffit de retrouver ce lien parmi les arêtes incidentes à s_i avant de le désactiver,
- si $Ind_c(s_i) = 0$, il suffit de tester si une des arêtes négatives (de s_i) est liée à son cycle. Si c'est le cas il faut remplacer ce lien par l'une des arêtes positives substituées.
- si $Ind_c(s_i) < 0$, il faut éventuellement parcourir tout le cycle \mathcal{C}^i contenant les arêtes négatives pour retrouver ce lien et le changer avec l'une des arêtes positives de la première chaîne rencontrée.

2. on réalise le contour trouvé en parcourant chacun de ses cycles.

Le pré-calcul de l'indice et des arêtes incidentes à un sommet n'est pas vraiment utile mais il facilite la lecture de l'algorithme sans en augmenter la complexité.

Les structures de données : d'après notre présentation il est clair que la structure de données du polyèdre doit mettre en valeur ses arêtes. Nous choisissons une structure d'arêtes ailées ([Bau75]) comme sur la figure 3.20. Chaque arête a également un pointeur vers une

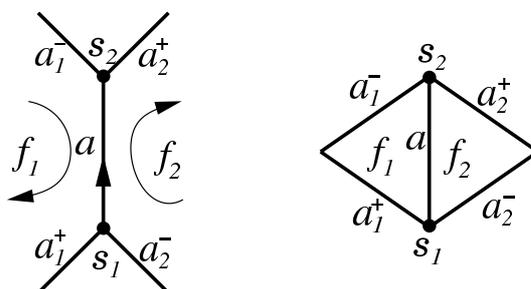


FIG. 3.20 – À gauche : la structure d'arête ailée pour un polyèdre orientable. À droite : le cas d'une triangulation.

arête suivante (resp. prédécesseur) pour l'éventuel cycle la contenant. On ajoute enfin un pointeur vers ce cycle. Nous avons ainsi la structure suivante pour une arête orientée (s_1, s_2) avec $c(s_1) < c(s_2)$:

SOMMET* s_1	SOMMET* s_2
ARETE* a_1^-	ARETE* a_2^-
ARETE* a_1^+	ARETE* a_2^+
ARETE* PRED	ARETE* SUIV
-	CYCLE* C

PRED, SUIV et le cycle courant C sont initialisés à NIL.

Un contour est une liste chaînée de cycles et un cycle est un pointeur vers l'une de ses arêtes. Cette arête pointe en retour sur son cycle.

Par ailleurs nous conservons un tableau des sommets ordonnés par la couleur. Chaque sommet pointe sur l'une quelconque des arêtes qui lui est incidente. La structure SOMMET est la suivante :

REEL x,y,z
REEL couleur
ARETE* a

Il est possible à partir de l'arête pointée par un sommet et de la structure de donnée en arêtes ailées de construire en temps et place $\mathcal{O}(k)$, la liste ordonnée circulairement dans le sens positif des k arêtes incidentes à ce sommet (voir [Bau75]). La structure de donnée AR_VOIS de la liste des arêtes voisines est

ARETE* a
AR_VOIS* prec
AR_VOIS* succ

Analyse de la complexité : pour chaque sommet nous devons

1. parcourir la liste de ses k_i arêtes incidentes pour faire les échanges de chaînes,
2. lorsque le sommet a un indice strictement négatif nous devons retrouver l'arête du cycle courant, de longueur l_i , qui fait le lien avec ce cycle,
3. pour réaliser le contour nous devons parcourir tous ses cycles d'arêtes ; nous notons L_i la longueur totale du contour courant.

Comme chaque arête est incidente à deux sommets nous avons

$$\sum_{i=1}^n k_i = 2 \# \text{ arêtes} = \mathcal{O}(n)$$

En notant m le nombre de sommet d'indices négatifs et

$$l = \max_i l_i \quad \text{et} \quad L = \max_i L_i$$

on a

$$\text{coût total} = \sum_{i=1}^n k_i + \sum_{\text{Ind}(s_i) < 0} l_i + \sum_{i=1}^n L_i = \mathcal{O}(n + ml + nL)$$

Dans le cas le pire nous pouvons avoir $\sum_{i=1}^n L_i = \mathcal{O}(n^2)$ comme sur l'exemple de la figure 3.21.

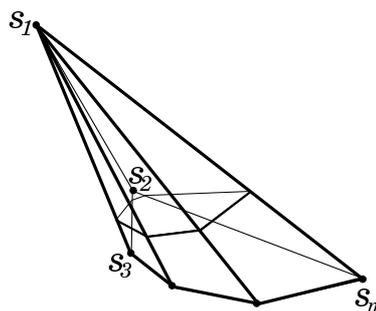


FIG. 3.21 – Un cône tronqué par un plan oblique et tel que $d(s_i) < d(s_{i+1})$.

3.4.2 Le cas pratique

Dans la pratique nous n'utilisons pas cet algorithme car :

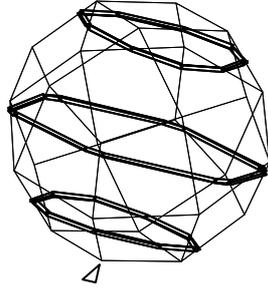
1. nous n'utilisons qu'un nombre donné à l'avance de lignes de niveau pour calculer l'axe de l'objet (ex : 100),
2. l'objet n'est pas nécessairement triangulé,
3. le graphe utilisé par l'algorithme de Dijkstra est le graphe du polyèdre complété par les cliques de chaque face,
4. On ne cherche pas à modifier la distance trouvée pour obtenir un coloriage.

Le premier point se justifie par le fait qu'un objet polyédrique est souvent défini par plusieurs centaines, voire plusieurs milliers de sommets et que nous n'avons pas besoin de tant de points pour définir une courbe.

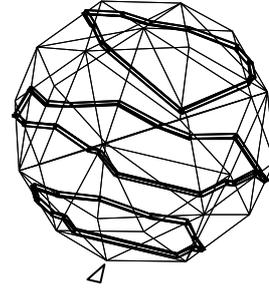
Le troisième point vient, comme nous l'avons déjà remarqué, du fait que la complétion du graphe du polyèdre par les cliques de ses faces fournit des distances plus proches des distances réelles lorsqu'on utilise l'algorithme de Dijkstra. Ceci apparaît clairement sur la figure 3.22. Sur cette figure l'objet est une sphère polygonalisée avec des quadrilatères. Les trois courbes de niveau correspondent respectivement à 25%, 50% et 75% de la distance maximales et sont représentées par des tubes fins.

Nous décrivons ici un algorithme plus adapté à notre besoin. Nous supposons que les faces du polyèdre, désormais noté \mathcal{P} , sont bien plongées, c'est à dire que deux sommets quelconques d'une même face n'ont pas les mêmes coordonnées dans \mathbb{R}^3 .

Par mesure de concision nous appelons *Dijkstra modifié* l'algorithme de Dijkstra appliqué au graphe complété par les cliques des faces. Nous appelons **liaison** une arête de ce graphe complété pour ne pas la confondre avec les arêtes de \mathcal{P} . Remarquons cependant que toute arête est une liaison.



Les faces du polyèdre sont des quadrilatères. Elles sont complétées par leurs deux diagonales non visibles sur la figure.



Le polyèdre est triangulé avant l'application de l'algorithme de Dijkstra.

FIG. 3.22 – Trois courbes de niveau. Le petit triangle indique le point source.

Enfin, nous notons d la fonction distance au point source sur P qui n'est plus nécessairement un coloriage.

Problème à résoudre : on cherche à construire p lignes de niveau équiréparties sur le polyèdre.

Principe de l'algorithme : on repère les maximums locaux et on détermine la plus grande distance notée d_M . Nous devons construire les p lignes de niveau $d_i = \frac{i}{p+1}d_M$ pour $i = 1, \dots, p$. Pour cela on redescend à partir de chaque maximum local le long de sa géodésique (donnée par Dijkstra modifié) jusqu'à rencontrer une liaison (σ_1, σ_2) telle que

$$d(\sigma_1) < d_i \leq d(\sigma_2) \quad (3.4)$$

La seconde inégalité de 3.4 n'est pas stricte car il se peut que d_i soit égale à la distance d'un ou plusieurs sommets. Nous pouvons cependant conserver stricte la première inégalité d'après l'hypothèse de bon plongement des faces.

Si la liaison (σ_1, σ_2) fait partie d'une clique d'une face f de \mathcal{P} , on en déduit une arête de f (et de \mathcal{P}) vérifiant 3.4 (c.f. figure 3.23). Il suffit pour cela de parcourir la liste ordonnée circulairement des arêtes de f , à partir d'une arête incidente à σ_1 , jusqu'à rencontrer une arête a_i^0 vérifiant 3.4. L'arête a_i^0 sert de point de départ à la construction d'un cycle de niveau d_i .

La construction d'un cycle fonctionne toujours sur le même principe :

pour une arête a vérifiant 3.4, l'orientation de a et de \mathcal{P} permet de discriminer sur chacune des deux faces incidentes une arête de sortie a_s et une arête d'entrée a_e comme sur

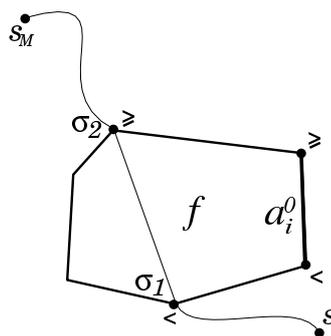


FIG. 3.23 – Le long de la géodésique d'un maximum local s_M on rencontre une face qui contient une arête du contour cherché.

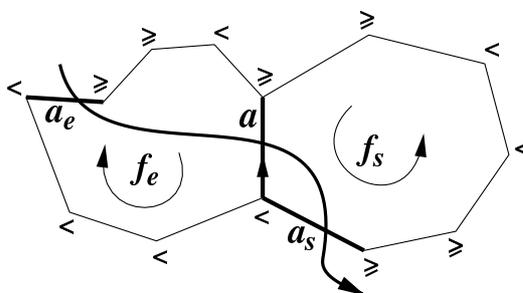


FIG. 3.24 – L'orientation de a et de \mathcal{P} permet de distinguer une face f_s et f_e .

la figure 3.24. Partant de l'arête a_i^0 on progresse par l'une de ses arêtes de sortie jusqu'à rencontrer à nouveau l'arête a_i^0 .

Remarque 3.4 Lorsque \mathcal{P} est une triangulation et d_i n'est pas égale à la distance d'un sommet ce procédé de descente à partir des maximums locaux permet d'obtenir **tous** les cycles de niveau d_i . En effet la proposition 3.5 reste vraie, que d soit un coloriage ou non. Pour chaque cycle de \mathcal{C}_{d_i} la géodésique associée à l'un de ses maximums internes le rencontre en une arête qui est celle choisie comme point de départ. Deux maximums locaux donnent ainsi lieu à deux cycles disjoints ou confondus.

3.5 Création d'axe

Nous avons montré comment obtenir des lignes de niveau sur un objet à partir de la distance géodésique à un point source. Ces lignes de niveau nous servent à définir des points

de l'axe de l'objet par moyennage. Puis nous interpolons cette liste de points (ordonnée) à l'aide d'une courbe spline.

Nous proposons à l'utilisateur quelques fonctionnalités pour modifier cet axe :

- au lieu de répartir les niveaux entre la source (de distance nulle) et le sommet le plus éloigné (de distance d_M), il peut spécifier deux pourcentages α et β de d_M entre lesquels répartir les lignes de niveau. Ceci permet d'agrandir les calottes aux extrémités de la partie cylindrique déterminée par l'axe.
- Il a également le choix entre plusieurs modes de moyennages des contours.
- Il peut désigner un sommet quelconque comme point source à l'aide de la souris.

Dans notre implémentation le moyennage d'une ligne de niveau peut être obtenu

- par son barycentre, c'est à dire comme le barycentre des milieux des segments qui la composent, pondérés par la longueur de ces segments,
- ou comme le point le plus proche des droites supports de chaque segment, pondérés par leur longueur.

Dans les deux cas, il s'agit d'une heuristique pour obtenir un point "centré" par rapport au contour.

3.5.1 Quelques résultats

Nous présentons différents calculs d'axes pour un cygne. Ce dernier est composé de 1728 facettes et est présenté vu de face, gauche et dessus dans la partie supérieure de la figure 3.25.

- Dans la partie inférieure de la figure 3.25, 20 lignes de niveau ont été construites, réparties entre 5% et 90% de la longueur maximale à partir du point source situé à l'extrémité du bec (repéré par le triangle blanc sur la figure 3.27). L'axe correspondant est représenté sous la forme d'un tube mince. Le moyennage des courbes de niveau est effectué à l'aide de leur barycentre. Les temps de calculs pour l'axe (et les courbes de niveau) sont de l'ordre de la fraction de seconde sur un machine SGI INDIGO2.
- La figure 3.26 illustre la bonne stabilité du calcul de l'axe en fonction du point source. Ici deux axes ont été calculés à partir de deux points sources indiqués par les triangles blanc et noir. On peut remarquer que les axes diffèrent sur le premier tiers de leur longueur et sont quasiment confondus dans la suite.

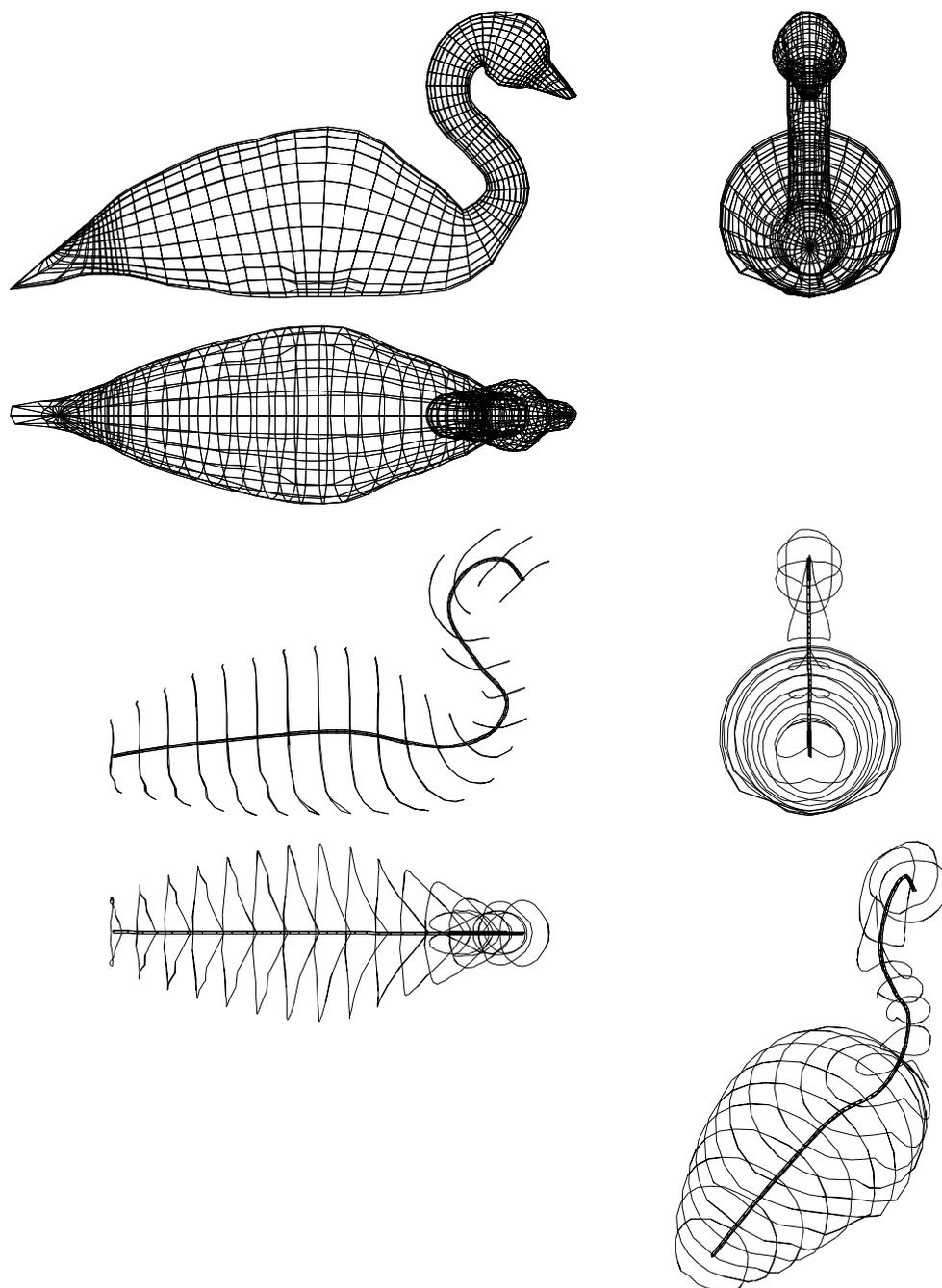


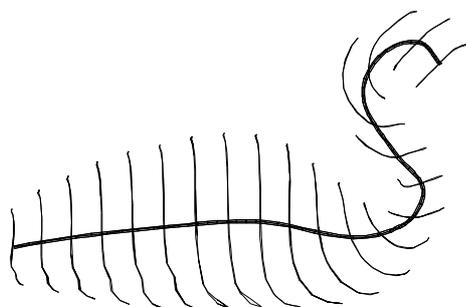
FIG. 3.25 – Un cygne, son axe et 20 courbes de niveau.

- Le calcul du centre des contours influe sur l'axe. Ceci est montré sur la figure 3.27.

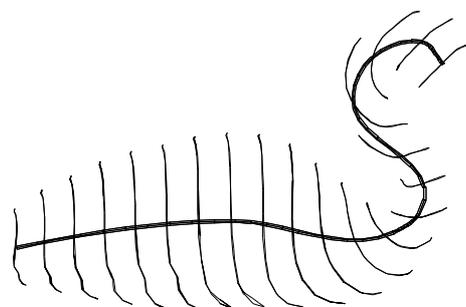
Le calcul des courbes de niveau ne dépend pas du plongement de l'objet mais seulement de son graphe et de la longueur de ses arêtes. Ceci peut s'observer sur la figure 3.28 où l'objet



FIG. 3.26 – À gauche : les flèches noires et blanches désignent les deux points sources utilisés pour le calcul de l'axe. Au milieu : les deux axes correspondant sont représentés sur la même figure. À droite : un détail de la figure du milieu.



Calcul de l'axe par moyennage avec le barycentre.



Calcul de l'axe par moyennage avec le point le plus proche des droites supports de chaque segments.

FIG. 3.27 – Deux calculs d'axe.

s'autointersecte plusieurs fois sans que cela crée de difficulté pour le calcul de l'axe.

Nous terminons cette section sur quelques exemples donnés figure 3.29...

3.5.2 À propos du centrage

L'axe que nous avons construit à l'aide des lignes de niveau ne répond pas à la condition d'étoilement qui permet le paramétrage voulu de la surface de l'objet. Comme précisé dans l'introduction de cette partie, nous n'avons pas réussi à construire un tel axe. Voici cependant quelques éléments de réponse pouvant ouvrir sur des voies de recherche.

D'après la forme de la paramétrisation recherchée (voir l'introduction de la partie II), l'axe doit vérifier deux conditions en chacun de ses points $\gamma(s)$:

1. une condition d'étoilement : le plan normal à $\gamma(s)$, noté \mathcal{P}_s , doit intersecter l'objet selon

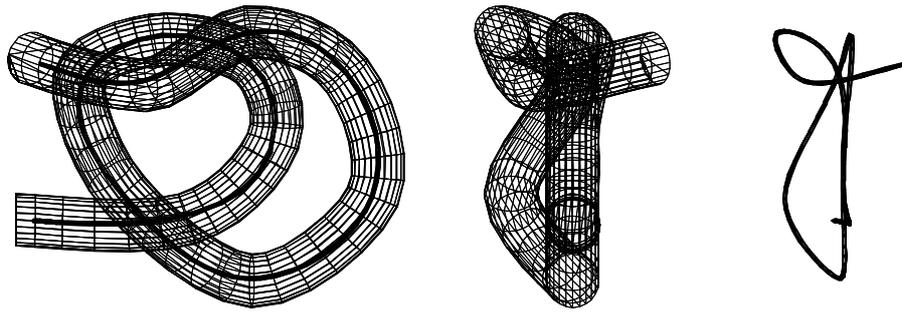
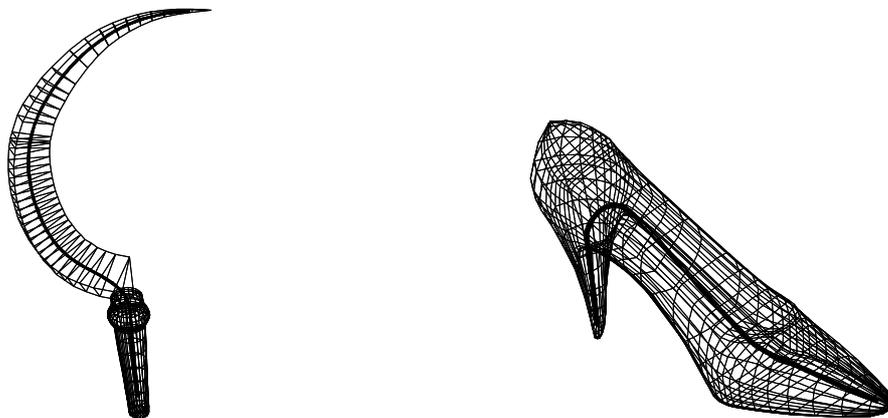


FIG. 3.28 – À gauche : une vue de face de l'objet et l'axe calculé en trait épais. Au milieu : une vue latérale de l'objet. À droite : la même vue de l'axe.



Une faucille et son axe

Un escarpin et son axe

FIG. 3.29 – Autres exemples de calcul d'axe.

une courbe telle que la composante $\mathcal{C}(s)$ contenant $\gamma(s)$ soit étoilée par rapport à ce point,

2. une condition de bon attachement (voir section 2.2 de la partie III) : la binormale en $\gamma(s)$ translatée au centre de courbure en $\gamma(s)$ sépare le plan normal \mathcal{P}_s en deux demi-plans dont l'un doit contenir la composante $\mathcal{C}(s)$ (voir figure 3.30 droite).

Nous allons exprimer ces deux conditions sous forme de l'annulation d'une certaine fonctionnelle.

Considérons tout d'abord la condition d'étoilement. Nous supposons pour cela que $\mathcal{C}(s)$ est une courbe fermée possédant partout une tangente⁸.

8. Une condition suffisante est que le plan normal \mathcal{P}_s intersecte transversalement l'objet. Cette condition

$\mathcal{C}(s)$ sépare \mathcal{P}_s en deux composantes connexes : l'intérieur qui est borné et l'extérieur de $\mathcal{C}(s)$. En chaque point p de $\mathcal{C}(s)$, nous notons \mathbf{n}_p la normale extérieure à $\mathcal{C}(s)$ (voir figure 3.30 gauche). La condition d'étoilement $\mathcal{C}(s)$ par rapport à $\gamma(s)$ se traduit ainsi par :

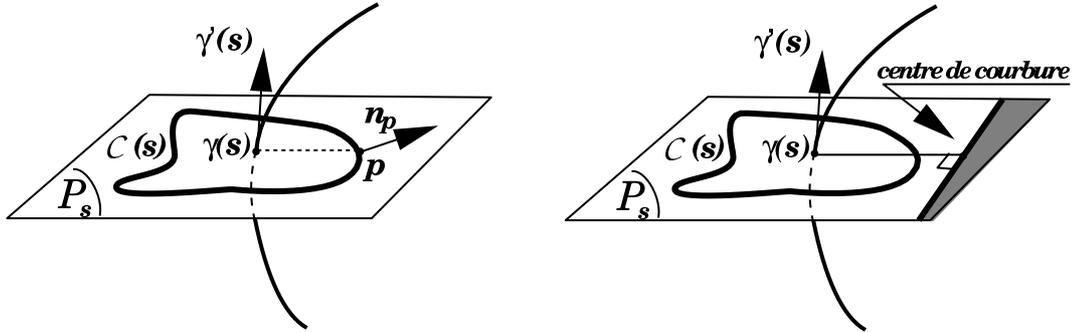


FIG. 3.30 – À gauche : la condition d'étoilement se traduit par $\langle \gamma(s)p, \mathbf{n}_p \rangle > 0$. À droite : la condition de régularité du paramétrage se traduit par la position de $\mathcal{C}(s)$ par rapport à la binormale translatée au centre de courbure.

$$\langle \gamma(s)p, \mathbf{n}_p \rangle > 0 \quad (3.5)$$

pour tout point p de $\mathcal{C}(s)$.

Soit alors $f_s : \mathcal{C}(s) \rightarrow \mathbb{R}_+$ une fonction intégrable telle que

$$\begin{cases} f_s(p) = 0 & \text{si } p \text{ vérifie (3.5)} \\ f_s(p) > 0 & \text{sinon} \end{cases}$$

alors $\mathcal{C}(s)$ est étoilé par rapport à $\gamma(s)$ si et seulement si

$$\int_{\mathcal{C}(s)} f_s = 0$$

Remarque 3.5 Nous pouvons par exemple choisir

$$f_{1,s}(p) = \langle \gamma(s)p, \mathbf{n}_p \rangle^2 \quad \text{si} \quad \langle \gamma(s)p, \mathbf{n}_p \rangle > 0$$

ou encore

$$f_{2,s}(p) = \|\gamma(s)p\|^2 \quad \text{si} \quad \langle \gamma(s)p, \mathbf{n}_p \rangle > 0$$

Nous avons

$$\int_{\mathcal{C}(s)} \langle \gamma(s)p, \mathbf{n}_p \rangle^2 \geq \int_{\mathcal{C}(s)} f_{1,s}$$

est clairement générique, c'est-à-dire vérifiée par "presque tous" les plans.

et

$$\int_{\mathcal{C}(s)} \|\gamma(s)p\|^2 \geq \int_{\mathcal{C}(s)} f_{2,s}$$

Les deux moyennages des lignes de niveau proposés précédemment (le point le plus proche des droites supports de chaque segment ou le barycentre) correspondent respectivement au minimum de ces deux intégrales lorsqu'on fait varier $\gamma(s)$. Notons par ailleurs que rechercher directement un centre d'étoilement des courbes de niveau n'a pas de sens car elles ne sont généralement pas planes.

La seconde condition de bon attachement traduit le fait que l'objet est localement paramétrable par γ . Elle sera vérifiée si

$$1 - \frac{1}{\|\gamma'(s)\|^2} \langle \gamma''(s), \gamma(s)p \rangle > 0 \quad (3.6)$$

pour tout point p de $\mathcal{C}(s)$.

Comme précédemment considérons $g_s : \mathcal{C}(s) \rightarrow \mathbb{R}_+$ une fonction intégrable telle que

$$\begin{cases} g_s(p) = 0 & \text{si } p \text{ vérifie (3.6)} \\ g_s(p) > 0 & \text{sinon} \end{cases}$$

alors $\mathcal{C}(s)$ vérifie la condition de paramétrisation locale si et seulement si

$$\int_{\mathcal{C}(s)} g_s = 0$$

Nous pouvons par exemple choisir

$$g_s(p) = \left(1 - \frac{1}{\|\gamma'(s)\|^2} \langle \gamma''(s), \gamma(s)p \rangle\right)^2 \quad \text{si} \quad \langle \gamma(s)p, \mathbf{n}_p \rangle > 0$$

Finalement l'axe γ sera considéré comme valide s'il vérifie

$$\int_{\gamma} \int_{\mathcal{C}(s)} f_s + \int_{\gamma} \int_{\mathcal{C}(s)} g_s = 0$$

et si l'union des $\mathcal{C}(s)$ recouvre l'objet éventuellement privé de deux calottes étoilées par rapport à chaque extrémité de γ .

En posant

$$F(s, \gamma, \gamma') = \int_{\mathcal{C}(s)} f_s \quad \text{et} \quad G(s, \gamma, \gamma', \gamma'') = \int_{\mathcal{C}(s)} g_s$$

l'équation précédente devient

$$\int_0^1 F(s, \gamma, \gamma') ds + \int_0^1 G(s, \gamma, \gamma', \gamma'') ds = 0 \quad (3.7)$$

Comme F et G ne sont pas négatives nous pouvons essayer de nous placer dans un contexte de calcul variationnel en cherchant un chemin γ qui réalise le minimum du membre de gauche de (3.7). Pour cela nous pouvons placer la recherche dans un espace de Sobolev du type $H^2([0, 1])$ (c.f. [Bre83]) en fixant les deux extrémités des chemins. La complexité des fonctionnelles F et G est cependant assez décourageante. On notera en particulier que le domaine d'intégration $\mathcal{C}(s)$ est fonction (implicite!) de $\gamma(s)$ et $\gamma'(s)$.

Une approche similaire a pourtant été proposée [SV90] pour rechercher une courbe squelette d'un contour plan. Ce cadre qui est largement plus simple puisque $\mathcal{C}(s)$ est réduit dans ce cas à deux points, donne lieu à une méthode de minimisation assez farfelue qui fait intervenir un calcul discret en parallèle sur une grille 2D où chaque arête est associée à un processeur.

3.6 Conclusion

Dans ce chapitre nous avons défini une méthode originale pour associer une structure axiale à un objet polyédrique. Pour cela nous avons en quelque sorte parcouru un chemin inverse à celui emprunté pour la reconstruction de surface à partir de contours ([CS78, Boi88, MSS92]) puisque nous avons obtenu des contours à partir d'une surface connue.

Les principales étapes pour la construction de l'axe d'un objet sont :

1. la désignation interactive d'un point source sur l'objet,
2. le calcul des distances géodésiques à ce point source par l'algorithme de Dijkstra,
3. le calcul et moyennage des cercles géodésiques à l'aide de l'algorithme exposé dans la section 3.4.2.

Les problèmes liés à la détermination de contours nous ont menés à développer une analogie combinatoire avec la théorie des fonctions de Morse. Les concepts essentiels pour mener à bien cette analogie sont ceux de triangulation, ou plus généralement de complexe simplicial, et de coloriage à valeurs dans un ensemble ordonné. Ceci nous a permis de mettre en avant le calcul des lignes de niveau sous forme de cycles d'arêtes, ce qui n'a pas été envisagé dans les algorithmes de visualisation où les lignes de niveau semblent néanmoins être importantes.

Notons par ailleurs que nous pouvons adapter cette construction d'axe à une surface implicite. Pour cela, nous obtenons un échantillonnage de la surface implicite avec un système de particules [WH94] (une autre approche du type [DTG95] suffit quand les surfaces implicites sont générées par des squelettes). Puis nous construisons un graphe sur ces points échantillonnés en reliant chaque point à ces k (par exemple $k = 4$) plus proches voisins comme dans [Ped95]. On applique alors l'algorithme de Dijkstra à partir d'un point source

sur ce graphe et on en déduit des ensembles de points équidistants de la source. Comme ce graphe n'est pas un polyèdre, ces ensembles de points ne sont pas structurés en cycles. On peut néanmoins les approximer par des courbes par des méthodes analogues à [Rou93] avant de moyenner ces courbes.

Par rapport à la notion classique d'axe médian notre méthode a l'avantage de fournir ipso facto une structure axiale. Cette méthode nécessite la distinction d'un point sur l'objet et la connaissance d'une métrique par opposition au plongement requis pour construire l'axe médian. Elle semble par ailleurs d'une grande stabilité par rapport aux objets pointés.

Il reste cependant que l'axe obtenu ne permet pas, dans le cas général de paramétrer l'objet comme voulu pour les métamorphoses. Une amélioration possible consiste à recentrer l'axe par des méthodes de type variationnelles où les contraintes d'étoilement et de courbure (c.f. section 3.5.2) sont exprimées sous forme de fonction à minimiser.

Troisième partie

Métamorphoses

Dans cette troisième et dernière partie nous présentons une technique de métamorphose. Cette technique s'applique à des objets de forme cylindrique qui sont *étoilés par rapport à une courbe*. Cette courbe sert à paramétrer et déformer les objets. Le paramétrage induit naturellement une correspondance entre les objets, et la déformation est obtenue par transformation des courbes associées à chaque objet.

Dans la première partie, nous avons montré comment effectuer cette transformation grâce à une homotopie régulière.

Dans la seconde partie nous avons fourni un outil pour obtenir une courbe “centrée” à l'intérieur d'un objet. Par la suite nous supposerons que les objets à métamorphoser sont munis d'une courbe convenable pour la condition d'étoilement.

Avant d'explicitier notre algorithme de métamorphose, nous passerons en revue les diverses méthodes qui ont été proposées en infographie; puis nous positionnerons notre approche par rapport à ces méthodes.

Chapitre 1

État de l'art

Nous présentons les principales techniques de métamorphoses proposées en infographie. Nous distinguons deux familles d'algorithmes selon la (co)dimension des données à métamorphoser :

1. l'approche **volumique**, qui regroupe les métamorphoses d'images discrétisées (i.e. constituées de pixels) ou de volumes discrétisés (i.e. constitués de voxels),
2. l'approche **surfactive**, qui désigne les métamorphoses de courbes planes ou de surfaces de \mathbb{R}^3 .

1.1 L'approche volumique

1.1.1 Métamorphoses bidimensionnelles

La méthode la plus efficace pour transformer des images pixels est sans doute celle décrite par T. Beier et S. Neely dans [BN92]. Une image pixels est un quadrillage rectangulaire dont chaque rectangle, ou pixel, possède une couleur uniforme. On peut considérer une image pixels comme un échantillonnage d'une application I de \mathbb{R}^2 dans l'espace des couleurs (c.f. figure 1.1). La couleur étant codée par trois nombres représentant des pourcentages de lumière rouge, verte et bleue, une image apparaît ainsi comme le graphe de l'application I de \mathbb{R}^2 dans \mathbb{R}^3 . En ces termes la transformation d'une image s'interprète comme une interpolation de graphe. Une simple composition "verticale" de la forme

$$I_t(x, y) = C(I_0(x, y), I_1(x, y), t)$$

fournit dans la plupart des cas une image difficile à interpréter sur le plan sémantique (sic).

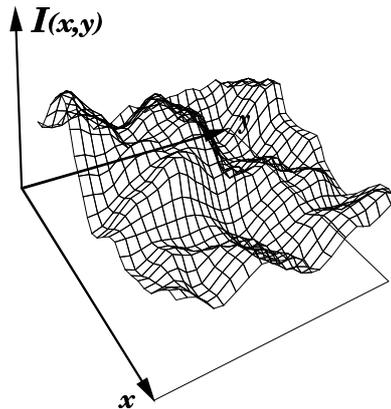


FIG. 1.1 – Le graphe associé à une image. Une seule composante de la couleur est représentée.

L'idée développée dans [BN92] est de se limiter à des images présentant les mêmes caractéristiques picturales. Intuitivement cette similarité exprime la possibilité de mettre en correspondance des traits caractéristiques (dans leur cas, des segments de droites) dans chacune des deux images. C'est le cas par exemple de deux images présentant chacune un visage vu sous le même angle, disons de face, puisqu'on peut mettre en correspondance des traits soulignant le nez, la bouche, les bords du visage...

Après une mise en correspondance des traits caractéristiques de deux images, les auteurs déforment le plan image xy de la première image (resp. la seconde) afin de surperposer ses traits caractéristiques avec ceux de la seconde image (resp. de la première image).

Après cette phase de déformation, qui porte le joli nom de *warping* en anglais, l'image interpolée est obtenue par composition des images déformées. Nous notons W_0 et W_1 les déformations respectives des deux images I_0 et I_1 (c.f. figure 1.2). Dans le cas idéal W_0 et W_1 sont inversibles et inverses l'une de l'autre. En s'inspirant du formalisme de [LCHS94] on a ainsi :

$$I_t(x, y) = C(I_0 \circ W_{t,0}(x, y), I_1 \circ W_{t,1}(x, y), t)$$

où $W_{t,0}$ est une homotopie entre l'identité et W_0 et où $W_{t,1}$ est une homotopie entre W_1 et l'identité. Dans [BN92] ces homotopies et la fonction C de composition sont des interpolations linéaires et les déformations W_0 et W_1 sont des sommes pondérées de déformations associées à chaque couple de traits caractéristiques.

Ces sommes pondérées ne respectent pas de façon précise la correspondance des traits caractéristiques. Pour remédier à ce défaut, S-Y Lee et al. [LCHS94] utilisent des modèles de surfaces déformables pour mettre en correspondance des points caractéristiques (au lieu de

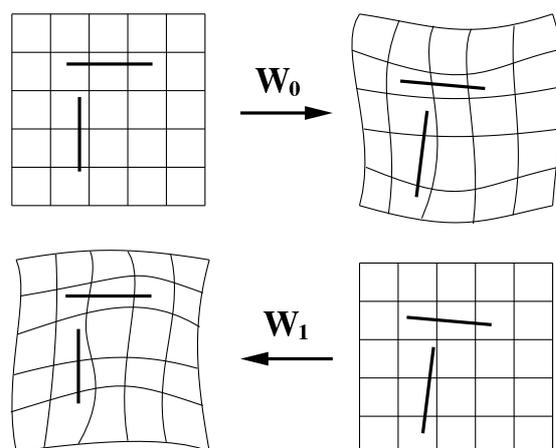


FIG. 1.2 – Les traits caractéristiques sont constitués de deux segments. Les déformations W_0 et W_1 ne sont pas nécessairement inverses l'une de l'autre.

segments comme dans [BN92]). Les auteurs permettent de plus à l'utilisateur, de spécifier un profil de transformation en certains points de l'image. Ceci permet d'effectuer des transitions à différentes vitesses dans l'image. Dans [LCHS94] on présente ainsi une transformation entre le visage d'une personne et la tête d'un chat où la partie gauche du visage est quasiment transformée en chat, tandis que celle de droite est encore peu déformée.

Dans [LCSW95] cette technique est encore améliorée : les traits caractéristiques ne sont plus simplement des points ou des segments de droites mais également des courbes (snakes) et les fonctions de déformation (warping) W_0 et W_1 sont calculées à partir de FFD [SP86] de manière à être bijectives.

Le résultat de ce type de métamorphose dépend en grande partie du choix et du positionnement des traits caractéristiques. Lorsque ce choix est judicieux cette méthode s'avère très efficace comme le montre son large emploi dans les vidéo-clips publicitaires. Il ne faut pas oublier cependant qu'il est essentiel, pour le bon fonctionnement de la méthode, que les deux images à métamorphoser aient des traits caractéristiques disposés de façon similaire.

1.1.2 Métamorphoses tridimensionnelles

Plusieurs méthodes ont été proposées pour métamorphoser des volumes de \mathbb{R}^3 . Elles diffèrent beaucoup dans leur principe, et peuvent reposer sur une très grande interaction avec l'utilisateur comme dans [LGL95], ou au contraire ne nécessiter aucune spécification comme dans [Hug92] ou [KR91].

L'attrait principal et commun à toutes ces méthodes réside dans la résolution simple, en apparence, des changements de topologie au cours des transformations.

- En substituant le mot “voxel” à celui de “pixel” A. Lerios et al. [LGL95] ont étendu la méthode de [BN92]. Ici les traits caractéristiques ne sont plus simplement des segments de droites mais également des points, des rectangles et des parallélépipèdes rectangles. Cependant cette extension est moins opportune qu'elle ne le semble au premier abord car :

1. le nombre de traits caractéristiques à positionner et mettre en correspondance est tel que l'aspect interactif devient excessivement lourd,
2. la visualisation d'un volume discrétisé par le lancer de rayon nécessite la définition d'une opacité pour chaque voxel, en plus de sa couleur. En discrétisant une partie de l'espace contenant un objet solide opaque les voxels ont, logiquement, une opacité nulle à l'extérieur de l'objet, égale à 1 à l'intérieur de l'objet et comprise entre 0 et 1 si le voxel intersecte la frontière de l'objet. En interpolant deux volumes discrétisés on obtiendra donc, sauf coïncidence exceptionnelle, un grand nombre de voxels ayant une opacité comprise entre 0 et 1. Il en résulte une zone de flou dans l'image reconstituée qui ne permet pas de distinguer un objet précis.

Cette technique s'apparente donc plus à une technique de mélange d'images qu'à une technique de métamorphose à proprement parler.

- J.F. Hughes [Hug92] propose une autre méthode de métamorphose de volume discrétisé à l'aide de la transformée de Fourier. Ici les objets sont vus comme des isosurfaces d'une fonction définie sur une partie de \mathbb{R}^3 (ou plus exactement \mathbb{Z}^3). En notant \mathcal{F} l'opération de transformée de Fourier, g_0 et g_1 deux fonctions définissant des objets, alors la fonction g_t définissant l'objet interpolé est donnée par :

$$g_t = \mathcal{F}^{-1}(s(t, \omega) \mathcal{F}(g_0)(\omega) + (1 - s(t, \omega)) \mathcal{F}(g_1)(\omega))$$

où ω représente un point du domaine fréquentiel et où $s(t, \omega)$ est une fonction comprise entre 0 et 1 qui dépend de la norme de la fréquence.

L'intérêt du calcul dans le domaine fréquentiel provient d'une séparation possible entre les basses fréquences et les hautes fréquences dans le schéma d'interpolation $s(t, \omega)$. Néanmoins, le lien entre les composantes fréquentielles et la géométrie des objets est loin d'être évident, les calculs sont prohibitifs ($O(n^3 \log n)$ où n est le nombre de voxels d'un côté du volume discrétisé) et les résultats (i.e. les images fournies dans [Hug92]) semblent indiquer des problèmes numériques importants.

Dans cette approche les objets peuvent avoir des topologies différentes. Plus exactement les objets étant considérés comme des ensembles isovaleur de fonctions, ce sont ces fonctions qui sont interpolées et non les surfaces isovaleurs. Néanmoins les ensembles isovaleurs d'une fonction différentiable ne sont pas toujours des surfaces. Une condition suffisante est que la fonction soit une submersion sur l'ensemble isovaleur considéré. Cette condition est rarement prise en compte et son non respect peut générer des incohérences telles que l'apparition d'arêtes vives alors que les objets à métamorphoser sont parfaitement lisses ou encore l'apparition de faces ou d'arêtes pendantes...

Notons encore un problème posé par l'utilisation des volumes discrétisés : le calcul d'une image sur l'écran de l'ordinateur à partir de données voxels s'obtient usuellement par des techniques de lancé de rayons ou de polygonalisation. L'instabilité de ces techniques au cours du temps génère des effets de scintillements sur les contours des images et nécessite des techniques d'anti-crénelage temporel.

- Une autre approche volumique utilise la somme de Minkowski [KR91]. Le traitement proposé par A. Kaul et J. Rossignac dans [KR91] est surfacique mais il s'agit avant tout de combiner deux volumes par la somme de Minkowski.

Rappelons que la somme de Minkowski de deux ensembles A et B d'un espace vectoriel est l'ensemble

$$A \oplus B = \{a + b \mid a \in A \text{ et } b \in B\}$$

En notant $t.A$ l'homothétie de rapport t appliquée à A , l'interpolation de deux objets A_0 et A_1 est obtenue par

$$A_t = (1 - t).A_0 \oplus t.A_1$$

On peut facilement vérifier que la frontière de la somme (de Minkowski) de deux objets fermés est incluse dans la frontière de la somme des frontières des objets. De ce fait on peut réaliser une interpolation basée sur les frontières des objets. A. Kaul et J. Rossignac proposent une interpolation de deux objets polyédriques A et B . Leur algorithme se fonde sur la propriété suivante :

les faces F_t du polyèdre interpolé à l'instant t sont nécessairement de la forme

$$\begin{aligned} F_t &= (1 - t).s_A \oplus t.f_B \\ \text{ou } F_t &= (1 - t).f_A \oplus t.s_B \\ \text{ou } F_t &= (1 - t).a_A \oplus t.a_B \end{aligned}$$

s , a et f désignant respectivement des sommets, arêtes ou faces des polyèdres (c.f. figure 1.3). Un critère sur la normale de F_t permet en plus de restreindre l'ensemble

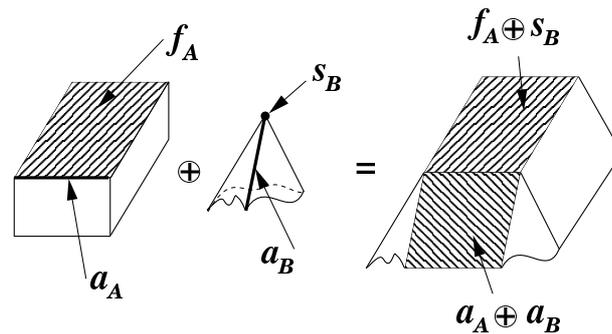


FIG. 1.3 – Différents types de faces dans la somme de Minkowski de deux polyèdres A et B (figure extraite de [KR91]).

des faces possibles.

La liste des faces F_t peut être construite en temps et place $O(n^2)$ si les polyèdres ont $O(n)$ faces. L'algorithme est amélioré lorsque les polyèdres sont convexes. L'animation est ensuite obtenue en affichant à tout instant les faces F_t par une technique de tampon de profondeur (z-buffer) car les faces F_t qui ne sont pas sur la frontière sont nécessairement à l'intérieur de l'objet interpolé.

Le principe des sommes de Minkowski est simple et se prête bien à un traitement algorithmique comme il est montré dans [KR91]. Il s'agit, à ma connaissance, de la seule méthode proposée capable de traiter des objets de forme et de topologie quelconque. Cette généralité s'obtient cependant au détriment du contrôle des déformations. En effet, ni l'échelle, ni la position relative des objets n'influent sur la forme des métamorphoses. Seule l'orientation relative des objets permet de les modifier.

Notons tout de même qu'il est possible de combiner des opérations d'homothétie et de somme de Minkowski pour produire des splines dont les points de contrôle sont des objets polyédriques [RK94]. Ceci permet effectivement de varier les transformations possibles entre deux objets, mais la complexité devient rapidement très importante ($O(n^4)$ pour une spline de degré 3).

1.2 L'approche surfacique

Nous incluons dans cette catégorie les transformations de courbes polygonales planes et de surfaces polyédriques de \mathbb{R}^3 . Les travaux existants séparent le problème en deux parties :

1. **La mise en correspondance** : on établit une bijection entre les surfaces des deux objets à métamorphoser.

2. **L'interpolation** proprement dite : on définit la trajectoire (comprise entre deux sommets mis en correspondance) de chaque sommet de l'objet interpolé.

Ci-dessous, nous détaillons séparément les approches spécifiques aux courbes et aux surfaces.

1.2.1 Métamorphoses de contours

Le problème des métamorphoses de formes polygonales est assez proche de celui de la reconstruction de surface à partir de coupes planes [Boi88] et les deux applications se font souvent référence comme dans [CP89]. En effet, dans le cadre de la reconstruction de surface à partir de coupes, il faut relier (et donc mettre en correspondance) les points de chaque coupe pour reconstruire la surface. Les techniques de mise en correspondance font souvent appel au produit des graphes de chaque coupe et la recherche d'une correspondance se traduit par la recherche d'un chemin dans ce graphe, également appelé graphe torique (c.f. figure 1.4). En attribuant un coût aux arêtes de ce graphe on peut rechercher un chemin de coût

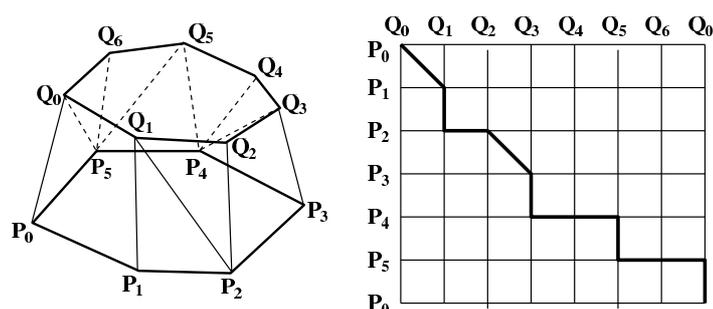


FIG. 1.4 – La mise en correspondance de deux contours P et Q peut être représentée par un graphe torique. Chaque point de ce graphe représente une arête de liaison entre les deux contours.

minimal et se ramener ainsi à un problème plus général [FKU77]. Dans [MSS92] les auteurs rappellent différentes expressions possibles de ces coûts et leur signification géométrique.

J.D. Boissonnat a proposé une approche volumique [Boi88] dans laquelle il reconstruit le volume séparant deux coupes à l'aide de la triangulation de Delaunay de l'ensemble formé par les sommets de ces coupes.

Une autre tendance consiste à appliquer chaque coupe sur une forme plus simple puis à utiliser ces formes simples pour la mise en correspondance. A. Ekoule et al. [EPO91] et S.E. Chen et R.E. Parent [CP89] utilisent par exemple une "projection" sur l'enveloppe convexe des coupes avant de mettre en correspondance ces enveloppes convexes.

En ce qui concerne les applications plus spécifiquement graphiques quelques approches ont été proposées :

- T.W. Sederberg et E. Greenwood [SG92] présentent une méthode de mise en correspondance automatique des sommets de 2 polygones simples. L'idée est d'associer les sommets des 2 polygones de sorte que le travail effectué pour "tordre" et "étirer" le premier polygone durant l'interpolation (supposée linéaire) soit minimal. L'expression du travail est issue d'un modèle physique de fil de fer extensible.

Le problème de l'interpolation qui fait suite est résolu dans [SGWM93] à partir d'une interpolation linéaire des longueurs des arêtes et des angles aux sommets. Comme cette interpolation ne permet pas d'obtenir un contour fermé, les auteurs modifient l'interpolation des longueurs par un procédé de minimisation sous la contrainte de fermeture du polygone interpolé. Comme il est écrit à la fin de l'introduction de [SGWM93] : "The solution presented here is a heuristic whose justification lies in the fact that it generally seems to work well"¹.

- M. Shapira et A. Rappoport [SR95] proposent de résoudre le problème de l'interpolation à l'aide d'une décomposition des polygones en sous-parties étoilées. La mise en correspondance est supposée résolue au préalable : chaque polygone a donc le même nombre de sommets qui se correspondent deux à deux. Les décompositions de chaque polygone doivent être compatibles au sens où (c.f. figure 1.5) :
 - les sous-parties étoilées doivent se correspondre deux à deux et cette correspondance doit respecter les adjacences entre sous-parties,
 - les sommets de deux sous-parties associées doivent être en bijection en respectant les adjacences entre sommets,
 - deux sous-parties associées doivent admettre un point d'étoilement qui s'exprime de la même manière comme barycentre de sommets en bijection dans les deux sous-parties.

L'algorithme pour réaliser cette décomposition commune n'est pas décrit dans [SR95] mais semble complexe. Les auteurs indiquent seulement une complexité en $O(n^4)$ si l'on n'ajoute pas de sommets aux polygones initiaux et en $O(n^2)$ si l'on se permet d'insérer $O(n^2)$ sommets dans les polygones initiaux.

Un squelette arborescent (en pointillé sur la figure 1.5) est ensuite construit à partir du graphe d'adjacence des sous-parties étoilées, des points d'étoilement de ces sous-parties

1. La solution présentée ici est une heuristique dont la justification réside dans le fait que cela semble généralement bien fonctionner.

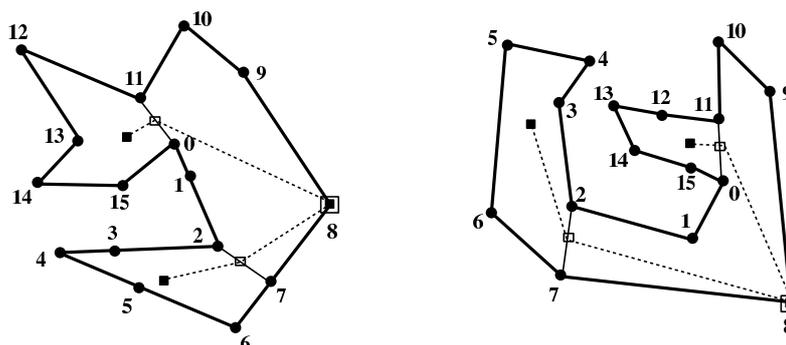


FIG. 1.5 – Deux décompositions compatibles de polygones. Les points \bullet représentent les sommets des polygones, les points carrés et noirs représentent les centres d'étoilement des sous-parties et les points \square représentent les milieux des arêtes d'adjacences entre deux sous-parties. Le point 8 est la racine de l'arbre associé aux sous-parties (figure extraite de [SR95].)

et des points milieu des segments incidents à deux sous-parties. L'interpolation est réalisée en interpolant ce squelette puis en interpolant les sommets des sous-parties relativement à un système de coordonnées propre à chaque sous-partie.

L'interpolation des sommets incidents à deux sous-parties (les sommets 0, 11, 2 et 7 sur la figure 1.5) peut ne pas coïncider selon la sous-partie considérée. Les auteurs utilisent une heuristique (i.e. le barycentre) pour moyenniser ces interpolations. En fait nous faisons face ici au très délicat problème des embranchements ou bifurcations qui ne possède pas de solution canonique valable pour toutes les configurations. Il s'agit sans doute du point le plus sensible de [SR95]. D'une manière générale il peut sembler commode d'effectuer des interpolations par morceaux, encore faut-il être capable de recoller les morceaux après interpolation !

- L'approche utilisée par E. Goldstein et C. Gotsman [GG95] pour résoudre l'interpolation rejoint celle d'A. Ekoule et al. [EPO91] pour la mise en correspondance : en effet, le principe est de déformer continûment chaque polygone en une forme commune et simple (un cercle). Les auteurs se servent ensuite des trajectoires de chaque sommet au cours de cette déformation pour calculer les positions des sommets à tout instant. Cette déformation peut être définie de manière "canonique" en faisant évoluer les contours (polygonaux) dans un champ de vecteurs qui est fonction de la courbure des contours (c.f. [KTZ92, ST94] et [CS93] pour une étude sur l'évolution des surfaces en fonction de la courbure).

Dans [GG95] la correspondance est supposée indépendante de cette déformation. De ce

fait la signification géométrique de l'interpolation n'est pas évidente. Les illustrations de [GG95] ne sont pas d'ailleurs très convaincantes...

- Citons pour mémoire des algorithmes très généraux comme la somme de Minkowski ou la méthode utilisée par B. Geiger pour connecter deux polygones simples [Gei93] (intuitivement on raccorde les deux polygones puis on gonfle l'un en rétractant l'autre). Du fait de leur généralité les interpolations fournies par ces algorithmes ne sont pas toujours intéressantes du point de vue des formes produites.
- Citons enfin un algorithme plus anecdotique pour métamorphoser des polygones dont les côtés se correspondent deux à deux et sont parallèles [GH94]. Le problème, issu de la géométrie algorithmique, est ici de transformer des polygones "parallèles" par des déformations "parallèles" (i.e. les polygones restent "parallèles" au cours de la transformation).

1.2.2 Métamorphoses de surfaces

La mise en correspondance des modèles polyédriques de surface est beaucoup plus ardue que pour des courbes polygonales. De ce fait, la plupart des méthodes proposées ne considèrent que ce problème, laissant de côté celui de l'interpolation.

Le principe commun à toutes les approches est de construire un sur-graphe contenant les graphes de chaque polyèdre (supposés homéomorphes à une sphère).

- E.W. Bethel et S.P. Uselton [BU89] transforment le problème de la mise en correspondance en un problème d'isomorphisme d'arbres (au sens de la théorie des graphes). Pour cela les auteurs considèrent le graphe dual de chaque polyèdre et le transforment en arbre par un parcours en largeur (c.f. [CLR94] §23.2) à partir d'une face spécifiée par l'utilisateur. Les arbres associés à chaque polyèdre sont alors mis en correspondance à partir de la racine, en ajoutant au fur et à mesure des sommets ou des faces dégénérées dans chaque graphe lorsqu'il n'y a pas correspondance.
- R.E. Parent [Par92] utilise un procédé dichotomique pour la mise en correspondance. Les polyèdres (de topologie sphérique) sont récursivement coupés en deux en parallèle afin d'associer les morceaux entre les deux polyèdres (c.f. figure 1.6). Le processus s'arrête lorsque les morceaux sont réduits à des faces. Au cours de la subdivision, des sommets ou arêtes peuvent être introduits dans chacun des graphes. L'algorithme est conçu pour ajouter le moins possible de sommets et d'arêtes. La procédure de subdivision n'est cependant pas explicitée et il est difficile de se rendre compte de l'efficacité

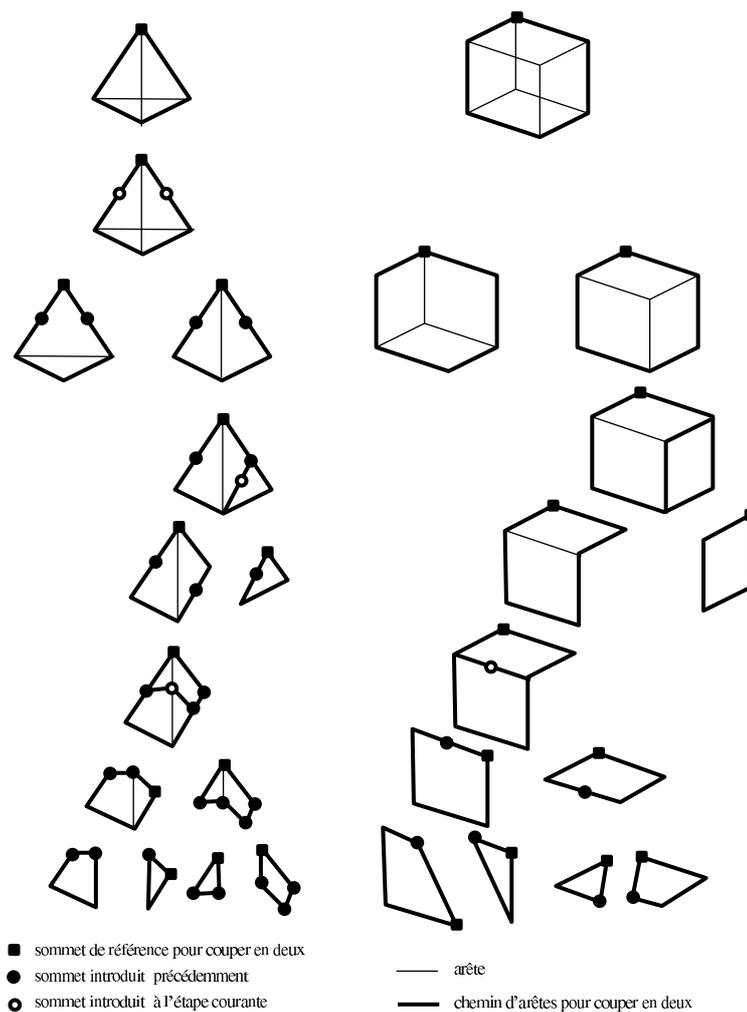


FIG. 1.6 – Subdivision récursive d'un tétraèdre et d'un cube. Seule la moitié de l'arbre de subdivision est montrée (figure extraite de [Par92]).

de la méthode.

Dans [BU89] comme dans [Par92], la mise en correspondance est indépendante du plongement des sommets et ne peut aboutir à des interpolations intéressantes sur le plan géométrique. En admettant qu'il soit judicieux de séparer la mise en correspondance de l'interpolation, il faut au moins définir des fonctions d'association qui soient "continues" par rapport au plongement des polyèdres, au sens où l'on s'intéresse autant à la proximité des sommets dans \mathbb{R}^3 qu'à leurs relations d'adjacence.

- L'approche la plus convaincante est celle décrite par J.R. Kent, R.E. Parent et W.E. Carlson dans [KPC91]. La qualité des interpolations produites s'obtient au dépend de la généralité : dans [KPC91] les polyèdres sont en effet supposés étoilés par rapport à un point.

Les auteurs construisent un sur-graphe en projetant radialement chaque polyèdre (à partir de son point d'étoilement) sur une sphère. La superposition des graphes projetés sur la sphère est calculée à l'aide d'un algorithme de découpe de polygones, du type Atherton et Weiler ([FDFH90], §19.1.4), en temps $O(n^2)$, les polyèdres ayant $O(n)$ arêtes. Cette réunion est ensuite reprojétée radialement sur chaque polyèdre (c.f figure 1.7). L'interpolation est linéaire et chaque sommet se déplace le long d'un rayon.

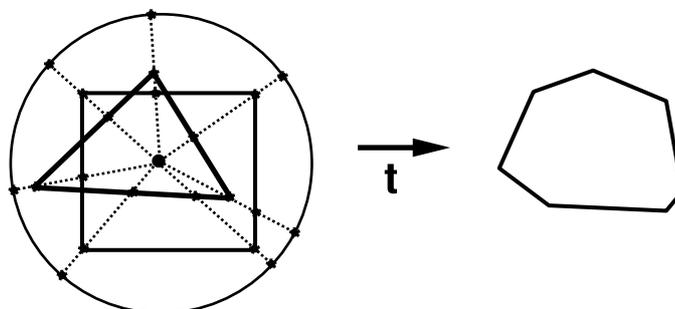


FIG. 1.7 – Principe (en deux dimensions) de l'interpolation entre un triangle et un carré.

L'orientation relative des objets et le choix du centre d'étoilement (dans le noyau de chaque polyèdre) sont les seuls paramètres de la métamorphose.

Dans [KCP92] on propose un catalogue d'extensions à des polyèdres non étoilés mais suffisamment particuliers pour être ramenés facilement à des sphères ou des convexes. Néanmoins le lien entre la géométrie des objets et l'interpolation, l'intérêt principal de [KPC91], est ici négligé. Par ailleurs l'algorithme de superposition des graphes est amélioré dans cet article.

1.2.3 Métamorphoses et évolutions de surfaces

Une approche très différente a été développée par X. Chen, P. Lienhardt et O. Terraz [CL92, Ter94, TL95] pour réaliser des métamorphoses. Dans ces travaux, le mot métamorphose doit cependant être compris dans le sens d'évolution plutôt que dans celui de passage d'une forme à une autre.

Cette approche repose essentiellement sur la structure topologique des objets qui s'identifient à des complexes cellulaires modélisés par des n-G-cartes [Lie93]. Les n-G-cartes repré-

sentent les relations d'adjacence des diverses cellules de l'objet (sommets, arêtes, faces...) au seul moyen de permutations sur un ensemble de "sommets" appelés *brins*. Les orbites des brins par ces permutations et leurs diverses composées permettent de retrouver les notions de sommet, arête, face, bord d'une surface, etc...

L'avantage de ce mode de représentation est de pouvoir spécifier des modifications (i.e. des évolutions) topologiques locales de façon simple et robuste. Les 2-G-cartes, qui modélisent des subdivisions de surface de dimension 2, sont plongées dans \mathbb{R}^3 par l'intermédiaire de leurs faces qui induisent les plongements des arêtes et sommets.

Le principe même des évolutions réside dans la *spécification* des règles de modification ou de croissance en fonction d'un temps (discret) d'horloge. Les règles de modification de plongement peuvent alors être liées au modèle de croissance.

1.3 Conclusion

La diversité des méthodes rencontrées et les fortes contraintes imposées aux objets dans chaque cas (transformation d'images similaires, de contours simples possédant le même nombre de sommets, de surfaces topologiquement équivalentes et étoilées...) montrent que la réalisation de métamorphose est un problème extrêmement complexe et loin d'être résolu dans tous les cas.

Si certaines méthodes sont a priori plus générales que d'autres, c'est au prix, soit d'un manque de contrôle sur la transformation ([KR91, Hug92]), soit d'une spécification totalement explicite demandée à l'utilisateur ([CL92, TL95]).

De fait, les méthodes qui produisent les meilleurs compromis entre le contrôle et l'automatisation sont celles qui se restreignent à des objets bien particuliers comme dans [BN92] ou [KPC91]. Dans ce cas il est possible de définir des transformations appropriées qui tiennent compte des hypothèses faites sur les objets à métamorphoser.

La démarche que nous avons suivie se rapproche beaucoup de celle proposée dans [KPC91] puisque nos métamorphoses s'appliquent à deux objets semblables (et simples) aussi bien sur le plan topologique que sur le plan géométrique. En effet, nos objets sont supposés être de topologie cylindrique (nous permettons cependant de fermer ce cylindre par des calottes, mais l'essentiel de la méthode réside dans le traitement de la partie cylindrique), et être *étoilés par rapport à un axe* (c.f. l'introduction de la partie II). La principale différence avec [KPC91] est donc d'avoir remplacé la condition d'étoilement par rapport à un point par celle d'étoilement par rapport à une courbe.

De plus, nous ne construisons pas de sur-graphe incluant les graphes des polyèdres d'origine. Mais nous échantillonons les polyèdres à l'aide d'une paramétrisation induite par leurs

axes.

L'utilisation d'un échantillonnage a plusieurs avantages :

- l'algorithme peut aisément être adapté à d'autres modèles de surfaces, telles que les surfaces implicites,
- la complexité de l'algorithme ne dépend essentiellement que de la phase d'échantillonnage,
- et il est possible de réaliser des métamorphoses entre un nombre quelconque d'objets sans augmenter la complexité, puisque la correspondance est implicitement déduite de l'échantillonnage.

Chapitre 2

Un algorithme de métamorphose

2.1 Schéma principal

Nous considérons deux objets surfaciques à métamorphoser. Nous supposons que chacun de ces objets est muni d'un axe¹ relativement auquel il est étoilé.

Chaque axe permet de décomposer l'objet associé en trois morceaux :

- deux calottes hémisphériques correspondant aux extrémités de l'axe,
- une partie cylindrique autour de l'axe.

Remarque 2.1 *La décomposition en trois parties est surtout utile pour ne pas avoir à définir un axe dont les extrémités adhèrent à la surface de l'objet. En fait, comme nous le verrons par la suite, les calottes et les parties cylindriques ont seulement, au cours de la métamorphose, un raccord continu même si ce raccord est C^∞ sur les objets clefs. En d'autres termes nous n'avons pas de raccord G^1 -continu².*

Dans la pratique, l'angle entre les plans tangents, de chaque côté des raccords, est faible, et le lissage induit par les algorithmes de "rendu" donne une impression de raccord G^1 . La possibilité d'avoir des calottes peut donc se justifier pour des applications à l'image de synthèse.

Les calottes et la partie cylindrique sont naturellement paramétrées par des systèmes de coordonnées sphériques ou cylindriques. En échantillonnant régulièrement l'espace des paramètres, nous construisons deux nouveaux graphes sur chaque objet, où chaque sommet

1. Notons que le mot axe est pour nous un synonyme d'arc compact, ce que nous avons encore appelé "courbe" dans la première partie de ce manuscrit. Par la suite nous utiliserons indifféremment ces vocables, et la notion de modèle de courbe (ou d'axe) se réfère à celle du chapitre 2 de la première partie.

2. On dit que deux morceaux de surfaces admettent un raccord G^1 -continu le long d'un bord commun si leurs plans tangents sont confondus le long de ce bord.

correspond à un paramètre échantillonné (c.f. figure 2.1). La mise en correspondance de ces

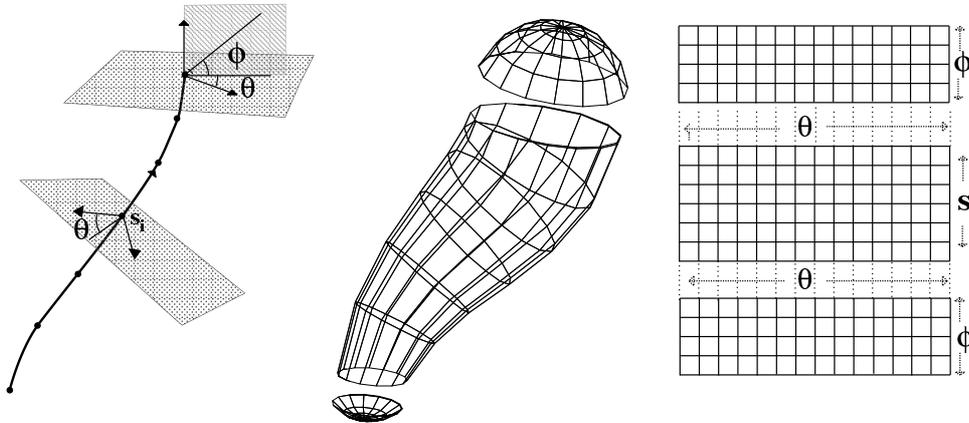


FIG. 2.1 – Un axe permet de décomposer un objet en trois parties.

nouveaux maillages est alors simplement obtenue en associant les sommets de mêmes paramètres.

L'interpolation des parties hémisphériques s'effectue ensuite radialement par rapport aux extrémités des axes, tandis que l'interpolation de la partie cylindrique est définie par des rayons orthogonaux aux axes.

Par ailleurs lorsque les objets comportent des éléments saillants tels que des arêtes vives, il est important de faire apparaître ces éléments dans les nouveaux maillages. Pour cela nous définissons une étape de recalage où nous déformons localement les maillages échantillonnés, de manière à les adapter sur les éléments saillants.

La description de l'algorithme s'organise de la manière suivante :

- dans la section 2.2 nous examinons comment définir un système de coordonnées cylindriques à partir d'une courbe,
- dans la section 2.3 nous détaillons la phase de paramétrisation et d'échantillonnage,
- puis dans la section 2.4 nous montrons comment recalibrer les maillages ainsi obtenus sur les éléments saillants des objets,
- et nous explicitons la mise en correspondance et l'interpolation des objets dans la section 2.5,
- nous concluons finalement par des exemples de métamorphoses.

Cet algorithme de métamorphose a par ailleurs fait l'objet d'une publication [LV94].

2.2 Système de coordonnées cylindriques

Nous examinons ci-dessous comment réaliser un système de coordonnées dans un voisinage de \mathbb{R}^3 à partir d'une courbe.

Étant donné une courbe γ plongée dans \mathbb{R}^3 , il est possible de définir, sur un voisinage "tubulaire" V_γ de cette courbe, un découpage en tranches de V_γ . Ces tranches sont constituées de morceaux de plans normaux à la courbe γ . Ce découpage constitue une fibration de V_γ de base γ et dont la projection π_γ est la projection orthogonale sur la courbe (c.f. figure 2.2). En munissant chaque fibre d'un repère, et en choisissant une paramétrisation de la courbe,

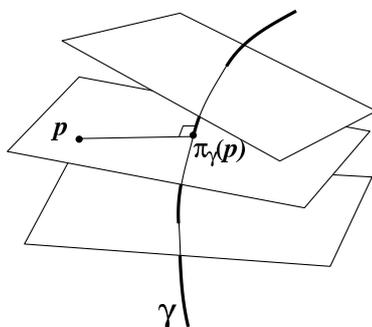


FIG. 2.2 – Le point p est projeté orthogonalement sur γ .

on peut définir un système de coordonnées :

$$\begin{aligned} V_\gamma &\rightarrow \mathbb{R}^3 \\ p &\mapsto (s, x, y) \end{aligned}$$

où s est le paramètre du point d'attache $\pi_\gamma(p)$ de p sur γ et (x, y) les coordonnées de p dans le repère associé à sa fibre.

Notons que le choix d'un repère dans chaque fibre revient au choix d'un repère mobile pour γ .

Dans [LCJ93, LCJ94] nous avons défini un outil de déformation pour l'image de synthèse fondé sur des systèmes de coordonnées associés à deux courbes γ_0 et γ_1 . L'idée est ici de "fixer" un objet par rapport à la courbe γ_0 et de lui faire subir la déformation induite par γ_1 . Un point p_0 est ainsi déformé en un point p_1 de sorte que les coordonnées de p_1 relativement à γ_1 soient identiques à celles de p_0 relativement à γ_0 , i.e. $(s_1, x_1, y_1) = (s_0, x_0, y_0)$.

Afin de déformer correctement un objet avec cette technique, il faut donc s'assurer que celui-ci est contenu dans un voisinage convenable de γ_0 puisque nous devons calculer les coordonnées de ses points relativement à γ_0 . En fait, la forme de ce voisinage dépend conjointement de la courbe et de l'objet comme on peut le voir sur la figure 2.3 : il n'est pas possible, à l'aide d'une information uniquement locale, de dire en quel point de l'axe un point de l'objet doit être attaché.

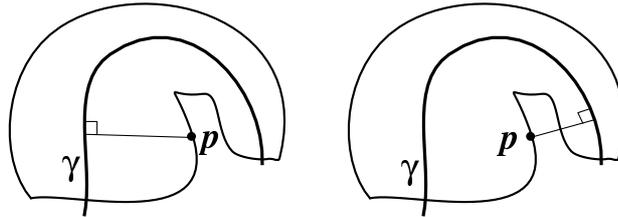


FIG. 2.3 – Le point p peut être projeté comme sur les deux figures, mais seul l'attachement de la figure de gauche peut aboutir à un attachement global de l'objet.

Dans [LCJ93] nous avons simplement attaché chaque point de l'objet au point le plus proche de l'axe, utilisant cette propriété d'extrémalité pour justifier le terme de projection orthogonale. Ce procédé ne permet donc pas de lever l'ambiguïté de la figure 2.3 et peut générer des incohérences lors de déformations axiales (c.f. figure 2.4).

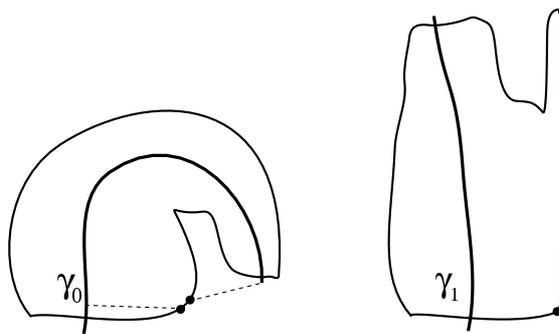


FIG. 2.4 – La fonction d'attachement est discontinue au niveau des points noirs. Une déformation de l'axe produit un objet discontinu.

Nous avons donc amélioré cette phase d'attachement de l'objet à l'axe. Nous avons de plus tenu compte du fait que l'axe est un modèle de courbe échantillonnée et non une courbe différentiable (voir le chapitre 2 de la première partie).

2.2.1 Système de coordonnées associé à un modèle de courbe

À partir d'une courbe modèle $\Gamma = (\Gamma_i, \mathbf{t}_i)$ munie d'un repère mobile (R_i) nous construisons une partition de l'espace entourant Γ à l'aide des plans de coupe Π_i définis par les points Γ_i et les vecteurs \mathbf{t}_i . La région comprise entre deux plans Π_i et Π_{i+1} forme un dièdre D_i dans lequel nous définissons un système de coordonnées. Afin de pouvoir définir correctement ces dièdres nous supposons désormais que les modèles de courbes vérifient les relations :

$$\langle \Gamma_i \Gamma_{i+1}, \mathbf{t}_i \rangle > 0 \quad \text{et} \quad \langle \Gamma_i \Gamma_{i+1}, \mathbf{t}_{i+1} \rangle < 0$$

Un point p est alors contenu dans un dièdre D_i s'il vérifie :

$$\langle \Gamma_i p, \mathbf{t}_i \rangle \geq 0 \quad \text{et} \quad \langle \Gamma_{i+1} p, \mathbf{t}_{i+1} \rangle < 0 \quad (2.1)$$

Le système de coordonnées à l'intérieur de D_i est défini à partir des points Γ_i et Γ_{i+1} et des repères R_i et R_{i+1} de la manière suivante :

pour chaque point p du dièdre D_i nous recherchons le plan Π_p , du faisceau engendré par Π_i et Π_{i+1} , qui contient p (c.f. figure 2.5). Ce plan admet une équation de la forme :

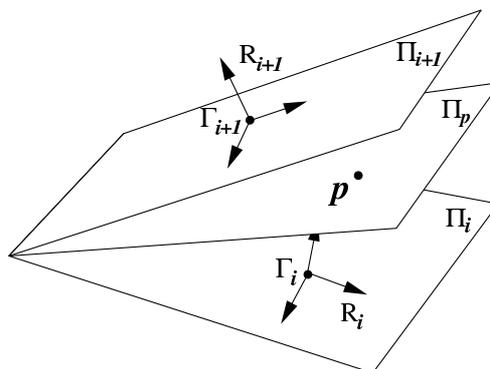


FIG. 2.5 – Le dièdre D_i .

$$(1 - \lambda_p) \langle \Gamma_i q, \mathbf{t}_i \rangle + \lambda_p \langle \Gamma_{i+1} q, \mathbf{t}_{i+1} \rangle = 0$$

où λ_p est obtenu en remplaçant q par le point p . La condition 2.1 assure que λ_p appartient à l'intervalle $[0, 1[$.

Puis nous munissons Π_p d'un repère de la manière suivante: le point Γ_i et le repère R_i sont appliqués sur Π_{i+1} par rotation autour de la droite $\Pi_i \cap \Pi_{i+1}$. Le point Γ'_i et le repère R'_i ainsi obtenus se déduisent respectivement de Γ_{i+1} et R_{i+1} à l'aide d'une translation et d'une rotation dans le plan Π_{i+1} (c.f. figure 2.6). Cette translation et l'angle de la rotation

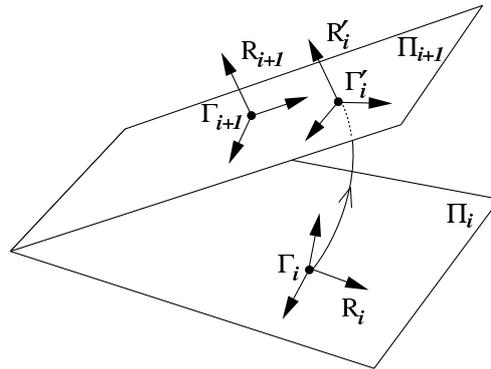


FIG. 2.6 – Report du système de coordonnées du plan Π_i dans le plan Π_{i+1} .

sont interpolés linéairement à l'aide du paramètre λ_p , le point et la rotation correspondante sont alors à nouveau transportés dans le plan Π_p par rotation autour de $\Pi_i \cap \Pi_{i+1}$.

Notons que lorsque le repère de Γ est le RTRP l'angle de rotation entre R'_i et R_{i+1} est toujours nul (c.f. section 2.2.2 de la première partie).

Définition 2.1 *La somme $i + \lambda_p$ et les composantes de p dans le repère de Π_p ainsi définies constituent les coordonnées de p relativement au dièdre D_i de la courbe Γ .*

2.2.2 Attachement d'un objet à un modèle de courbe

Comme dans le cas continu, il n'est pas possible de dire à quel dièdre un point doit être attaché sans tenir compte de la forme de l'objet auquel il appartient (c.f. figure 2.7). En fait, pour dire si un point de l'objet est bien attaché à l'axe nous devons définir ce qu'est un objet bien attaché.

Définition 2.2 *Un objet immergé dans \mathbb{R}^3 est attaché correctement par rapport à un modèle de courbe Γ si ses points peuvent être associés à des dièdres de Γ de sorte que leurs coordonnées dans ces dièdres, définies ci-dessus, varient continûment sur l'objet.*

Nous avons permis à l'objet d'être simplement immergé afin de pouvoir l'attacher s'il s'auto-intersecte. Ceci n'était pas possible en utilisant le procédé de [LCJ93]. De sorte qu'après déformation d'un objet avec un axe il n'était pas toujours possible de ré-attacher correctement l'objet déformé à l'axe déformé.

Savoir si un objet peut ou non être correctement attaché à une courbe peut être délicat, d'autant que cet attachement n'est pas nécessairement unique comme le montre la figure 2.8. Par contre, si nous savons attacher un point de l'objet à la courbe alors le reste de l'objet

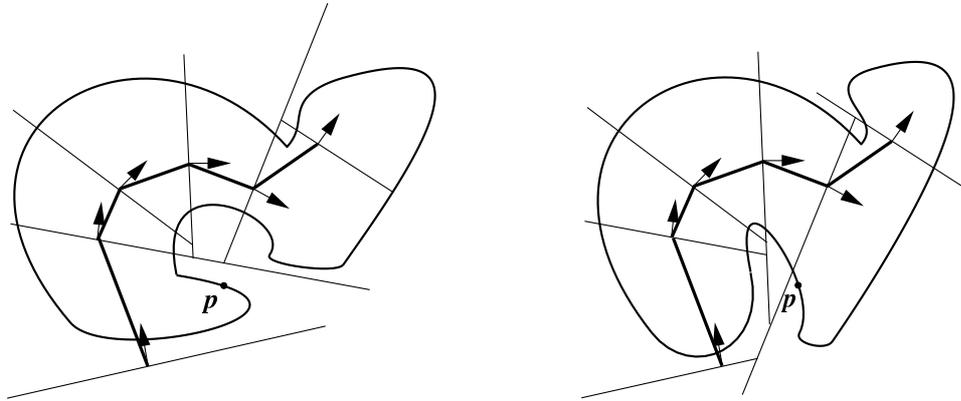


FIG. 2.7 – Le point p doit être attaché différemment selon l'objet auquel il appartient.

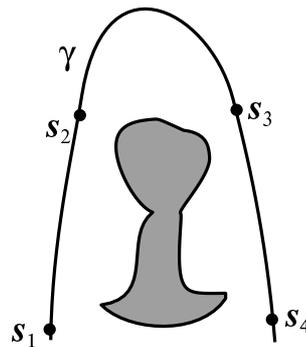


FIG. 2.8 – L'objet gris peut être attaché aussi bien sur la portion d'axe comprise entre s_1 et s_2 qu'entre celle comprise entre s_3 et s_4 .

peut être attaché par continuité. Nous utilisons ainsi l'algorithme suivant pour attacher les sommets d'un polyèdre :

Algorithme d'attachement Les données de l'algorithme sont constituées d'un polyèdre P , d'un modèle de courbe Γ et d'un sommet de P correctement attaché à Γ .

Par un parcours en largeur (c.f. [CLR94] §23.2) de P on attache progressivement les sommets. Pour cela on utilise le fait que si un sommet p est attaché au dièdre D_i , alors un sommet adjacent q est attaché comme suit :

- si $\langle \Gamma_i q, \mathbf{t}_i \rangle \geq 0$ alors q est attaché à D_j avec

$$j = \inf \{k \geq i \mid \langle \Gamma_k q, \mathbf{t}_k \rangle \geq 0 \text{ et } \langle \Gamma_{k+1} q, \mathbf{t}_{k+1} \rangle < 0\}$$

- sinon q est attaché à D_j avec

$$j = \sup \{k < i \mid \langle \Gamma_k q, \mathbf{t}_k \rangle \geq 0 \text{ et } \langle \Gamma_{k+1} q, \mathbf{t}_{k+1} \rangle < 0\}$$

Un argument de continuité permet en effet de voir que la coordonnée $i + \lambda$ est nécessairement monotone le long d'une arête.

Il est facile de voir que la complexité de l'algorithme d'attachement est proportionnelle à la somme du nombre d'arêtes et sommets de P et du nombre d'intersection des arêtes de P avec les morceaux de plans de coupe Π_i inclus dans le voisinage tubulaire concerné. Dans le pire des cas ce nombre est égal au produit du nombre d'arêtes de P par le nombre de sommets de Γ .

Remarque 2.2 *L'algorithme d'attachement requiert la donnée d'un sommet p de P correctement attaché à Γ . Dans notre cas, il n'existe qu'un seul attachement possible car l'axe est interne à l'objet. La sélection de ce sommet est obtenue en choisissant une demi-droite orthogonale à l'axe Γ en un point Γ_i , puis en choisissant p dans la face la plus proche de Γ_i qui intersecte cette demi-droite. Ce sommet peut ensuite être attaché facilement à Γ sans erreur (c.f. figure 2.9).*

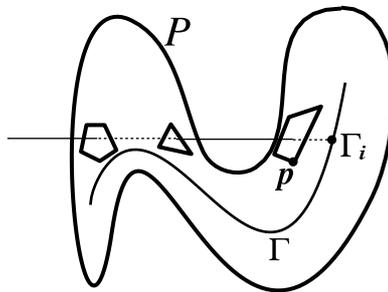


FIG. 2.9 – Le sommet p est attaché en progressant le long de Γ à partir du point Γ_i .

2.2.3 Extensions aux extrémités d'une courbe

Dans la mesure où les objets ne sont pas entièrement attachés à l'axe, mais où des parties sont également attachées aux extrémités, ce que nous appelons les calottes (c.f. figure 2.10), il est indispensable de s'assurer que le système de coordonnées le long de l'axe se prolonge convenablement dans les demi-espaces "au dessus" des extrémités. Nous allons voir que ce prolongement n'est possible que si la courbure de l'axe et la torsion du repère adapté sont nulles aux extrémités. Nous raisonnons sur l'extrémité $\gamma(1)$ pour ne pas avoir à inverser les

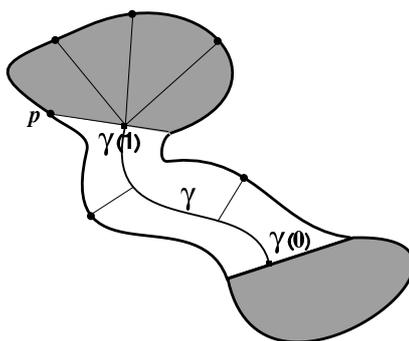


FIG. 2.10 – Les parties grisées de l'objet en trait épais sont attachées aux extrémités de l'axe γ .

repères entre l'axe et la calotte.

Comme nous l'avons vu au début de cette section, le système de coordonnées associé à l'axe est le suivant :

$$p \xrightarrow{\Phi} (s, x, y)$$

où s est le paramètre du point d'attache $\pi_\gamma(p)$ de p sur γ et (x, y) les coordonnées de p dans le repère associé à $\gamma(s)$. Plus précisément, si $R(s) = (\mathbf{e}_1(s), \mathbf{e}_2(s), \mathbf{e}_3(s))$ est le repère mobile, le long de γ , les coordonnées de p sont :

$$(\Phi_1(p), \Phi_2(p), \Phi_3(p)) = (s, \langle \gamma(s)p, \mathbf{e}_2(s) \rangle, \langle \gamma(s)p, \mathbf{e}_3(s) \rangle)$$

où s est une fonction de p déterminée implicitement par l'équation :

$$f(p, s) = \langle \gamma(s)p, \mathbf{e}_1(s) \rangle = 0$$

En effet, comme l'expression³ $\frac{\partial f}{\partial s} = -v(1 - \kappa \langle \gamma p, \mathbf{n} \rangle)$ n'est pas nulle, d'après l'hypothèse de bon attachement de l'objet, nous pouvons appliquer le théorème des fonctions implicites, et nous avons en particulier :

$$d\Phi_1(p) \cdot \mathbf{h} = ds(p) \cdot \mathbf{h} = -\frac{\partial_p f}{\partial_s f} \cdot \mathbf{h} = \frac{\langle \mathbf{h}, \mathbf{e}_1 \rangle}{v(1 - \kappa \langle \gamma p, \mathbf{n} \rangle)}$$

On en déduit :

$$d\Phi_2(p) \cdot \mathbf{h} = \langle \mathbf{h}, \mathbf{e}_2 \rangle + \langle \gamma p, \frac{d\mathbf{e}_2}{ds} \rangle ds(p) \cdot \mathbf{h}$$

et de même :

$$d\Phi_3(p) \cdot \mathbf{h} = \langle \mathbf{h}, \mathbf{e}_3 \rangle + \langle \gamma p, \frac{d\mathbf{e}_3}{ds} \rangle ds(p) \cdot \mathbf{h}$$

Par conséquent, pour prolonger le système de coordonnées cylindriques en une application linéaire au delà de l'extrémité $\gamma(1)$, il faut que la courbure κ soit nulle et que les produits

3. où $v = \|\gamma'\|$, et $\frac{d\mathbf{e}_1}{ds} = v\kappa\mathbf{n}$.

$\langle \gamma p, \frac{d\mathbf{e}_2}{ds} \rangle$ et $\langle \gamma p, \frac{d\mathbf{e}_3}{ds} \rangle$ soient nuls pour tout p attaché à une extrémité. Cette dernière condition est vérifiée si et seulement si le vecteur vitesse de rotation instantanée est normal à \mathbf{e}_1 . On vérifie assez simplement (c.f partie I, section 2.3.1) que cela équivaut à dire que l'angle entre le repère R et l'un quelconque des RTRP de γ a une dérivée nulle au niveau de l'extrémité.

Dans ces conditions nous avons pour tout point p du plan extrême associé à $\gamma(1)$:

$$d\Phi(p) \cdot \mathbf{h} = \left(\frac{1}{v} \langle \mathbf{h}, \mathbf{e}_1 \rangle, \langle \mathbf{h}, \mathbf{e}_2 \rangle, \langle \mathbf{h}, \mathbf{e}_3 \rangle \right)$$

Cette dernière expression est indépendante de p et peut donc être prolongée par une application linéaire dans la partie au dessus de $\gamma(1)$.

Nous avons ainsi réalisé un système de coordonnées de classe \mathcal{C}^1 . Avec ce système de coordonnées, nous pouvons donc redresser une surface régulière et obtenir une surface redressée également régulière (i.e. ayant partout un plan tangent).

Lorsque $v = 1$ aux extrémités, le redressement des calottes est une simple rotation, sinon il faut en plus effectuer une affinité de rapport $\frac{1}{v}$ perpendiculairement aux plans normaux aux extrémités.

2.3 Paramétrage et échantillonnage

Nous décrivons le processus d'échantillonnage d'un objet à partir d'un axe. En premier lieu, une paramétrisation de l'objet est définie, puis un échantillonnage de sa surface est construit. La surface est supposée être dans un voisinage convenable de la courbe (i.e. où l'on peut définir le système de coordonnées précédent).

2.3.1 Paramétrage à partir d'un axe

Pour paramétrer un objet surfacique relativement à un axe, il ne suffit pas de pouvoir l'attacher comme précédemment. En fait nous considérons qu'un objet peut être paramétré par un axe γ muni d'un repère mobile $R = (\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$ si (ayant confondu γ et sa paramétrisation):

1. la calotte attachée à l'extrémité inférieure $\gamma(0)$ admet une paramétrisation de la forme :

$$(\theta, \phi) \mapsto \gamma(0) + r_0(\theta, \phi) \mathbf{v}_0(\theta, \phi), \quad (\theta, \phi) \in [0, 2\pi[\times [0, \frac{\pi}{2}]$$

2. la partie centrale admet une paramétrisation de la forme :

$$(s, \theta) \mapsto \gamma(s) + r_1(s, \theta) \mathbf{u}(s, \theta), \quad (s, \theta) \in]0, 1[\times [0, 2\pi[$$

3. la calotte attachée à l'extrémité supérieure $\gamma(1)$ admet une paramétrisation de la forme :

$$(\theta, \phi) \mapsto \gamma(1) + r_2(\theta, \phi)\mathbf{v}_1(\theta, \phi), \quad (\theta, \phi) \in [0, 2\pi[\times [0, \frac{\pi}{2}]$$

où r_0 , r_1 et r_2 sont des fonctions strictement positives, et

$$\begin{aligned} \mathbf{v}_0(\theta, \phi) &= \cos \phi (\cos \theta \mathbf{e}_2(0) + \sin \theta \mathbf{e}_3(0)) - \sin \phi \mathbf{e}_1(0) \\ \mathbf{u}(s, \theta) &= \cos \theta \mathbf{e}_2(s) + \sin \theta \mathbf{e}_3(s) \\ \mathbf{v}_1(\theta, \phi) &= \cos \phi (\cos \theta \mathbf{e}_2(1) + \sin \theta \mathbf{e}_3(1)) + \sin \phi \mathbf{e}_1(1) \end{aligned}$$

La forme des paramétrisations montre que la partie centrale doit être ce qu'on appelle couramment un *cylindre généralisé* [Pon88] relativement à γ , tandis que les calottes doivent être étoilées par rapport aux extrémités respectives $\gamma(0)$ et $\gamma(1)$.

Inversement, si on se donne un axe à l'intérieur d'un objet, tel que :

1. sa surface peut être découpée en deux calottes et une partie centrale relatives à l'axe,
2. pour chaque point des calottes, le segment qui le lie à l'extrémité correspondante ne rencontre pas l'objet,
3. pour chaque point de la partie centrale, le segment qui le lie à sa projection sur l'axe ne rencontre pas l'objet,

alors, l'objet peut être paramétré par l'axe de la manière précédente. Par ailleurs, si l'objet est une surface régulière et que les droites support des segments considérés ci-dessus l'intersectent transversalement⁴ aux points considérés, alors la paramétrisation est régulière sur chacun des trois morceaux de la surface.

Dans la pratique nous employons des axes qui sont des modèles de courbes. Nous pouvons cependant conserver ces définitions et propriétés de paramétrage dans le cadre d'un modèle de courbe. Dans ce cas, seule la définition des points $\gamma(s)$ et des repères $(\mathbf{e}_1(s), \mathbf{e}_2(s), \mathbf{e}_3(s))$ doit être adaptée en fonction de la définition du système de coordonnées associé à un modèle de courbe (c.f. section précédente).

Le choix de cette paramétrisation est en fait uniquement lié à son caractère géométrique, au sens où le point de paramètre (s, θ) (resp. (θ, ϕ)) est obtenu en intersectant l'objet avec la demi-droite d'extrémité $\gamma(s)$ et de direction $\mathbf{u}(s, \theta)$ (resp. d'extrémité $\gamma(0)$ ou $\gamma(1)$ et

4. i.e. le plan tangent à la surface ne contient pas la droite support au point considéré.

de direction $\mathbf{v}_0(\theta, \phi)$ ou $\mathbf{v}_1(\theta, \phi)$). Par la suite nous appellerons ces demi-droites des **demi-droites d'échantillonnage**.

Nous insistons, par ailleurs, sur le fait que le procédé de métamorphose que nous proposons n'est pas essentiellement lié à cette paramétrisation mais à la propriété de fibration de l'objet *au dessus* d'une courbe immergée dans \mathbb{R}^3 .

2.3.2 Échantillonnage à partir d'un axe

Pour échantillonner un objet surfacique paramétré par un axe Γ , nous “projetons” une grille rectangulaire de l'espace des paramètres sur chacun des trois morceaux de surface. Les sommets de cette grille correspondent à un échantillonnage uniforme des paramètres s , θ et ϕ . Les nombres d'échantillons de chacun de ces paramètres sont spécifiés par l'utilisateur. Ils déterminent la finesse de l'échantillonnage. Si le nombre d'échantillons des paramètres s , θ et ϕ sont respectivement n , p et q , alors le nombre total d'échantillons sur l'objet est :

$$n_{tot} = p(n + 2q - 2).$$

Comme nous venons de le voir, un point de paramètre (s_i, θ_j) (resp. (θ_i, ϕ_j)) est obtenu par intersection de l'objet avec la demi-droite d'échantillonnage correspondante.

Jusqu'à présent nous n'avons pas explicité le modèle de représentation des objets surfaciques car il n'intervenait pas dans le paramétrage décrit. Par contre lors du calcul d'une intersection avec une demi-droite d'échantillonnage, nous tenons compte de ce modèle.

- Si l'objet est défini par des fonctions implicites, il suffit de calculer cette intersection, par un procédé de dichotomie, pour chaque couple de paramètres échantillons (s_i, θ_j) (resp. (θ_i, ϕ_j)).
- Si l'objet est défini par un maillage polyédrique, comme dans notre implémentation, cette intersection s'obtient en testant l'intersection de la demi-droite d'échantillonnage avec toutes les faces du polyèdre.

Afin d'éviter ces calculs, nous inversons le procédé en cherchant, pour chaque face de P , les échantillons contenus dans cette face. Pour cela, chaque face est “projetée” dans l'espace des paramètres (c.f. figure 2.11). Ceci nécessite en premier lieu de calculer les coordonnées des sommets de P relativement à l'axe, en utilisant l'algorithme d'attachement décrit plus haut. Puis, en tenant compte des coordonnées des sommets d'une face, nous pouvons retrouver facilement les échantillons intérieurs à cette face.

Nous détaillons ci-dessous la détermination des échantillons intérieurs à une face, selon que cette face est dans la partie cylindrique ou dans l'une des calottes.

Échantillonnage d'une face de la partie cylindrique Les échantillons contenus dans une face F de la partie cylindrique sont calculés par un balayage de la face par les plans

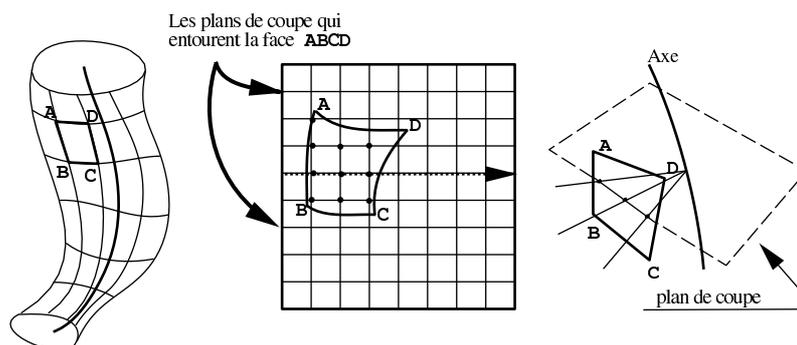


FIG. 2.11 – La face $F = ABCD$ est projetée dans l'espace des paramètres. Les échantillons intérieurs sont calculés par balayage à l'aide des plans de coupe.

de coupe Π_i . Seuls sont pris en compte les plans dont l'indice est compris entre les indices des dièdres associés à chaque sommet de F (c.f. figure 2.11). Pour chaque plan de coupe Π_i , on calcule le segment intersection $F \cap \Pi_i$ (les faces sont supposées planes et convexes). Les angles polaires θ_a et θ_b des extrémités a et b de ce segment sont calculés dans le repère (Γ_i, R_i) . On intersecte alors la face F avec les demi-droites d'échantillonnage de paramètre (s_i, θ_j) , où s_i correspond au plan Π_i et les échantillons θ_j sont compris entre les deux angles θ_a et θ_b (c.f. figure 2.12).

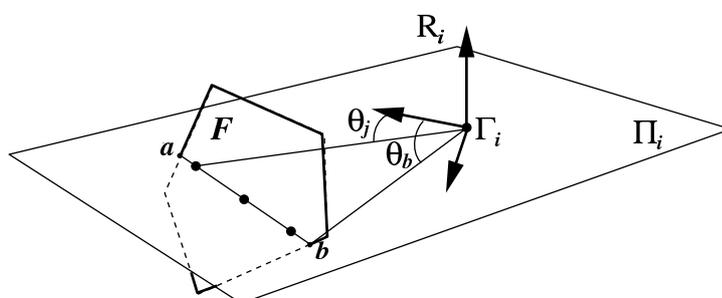


FIG. 2.12 – On détermine les échantillons dans l'intersection ab de la face F avec le plan de coupe Π_i .

Les échantillons calculés sont rangés au fur et à mesure dans un tableau rectangulaire représentant la grille d'échantillonnage de la partie cylindrique.

Échantillonnage d'une face d'une calotte La détermination des échantillons contenus dans une face F d'une calotte est en tout point similaire au cas précédent. Les plans de coupes Π_i , servant au balayage, sont ici remplacés par les quarts de plan, correspondant aux

angles θ_i , du faisceau engendré par la droite (Γ_0, t_0) (resp. (Γ_n, t_n)) (c.f. figure 2.13).

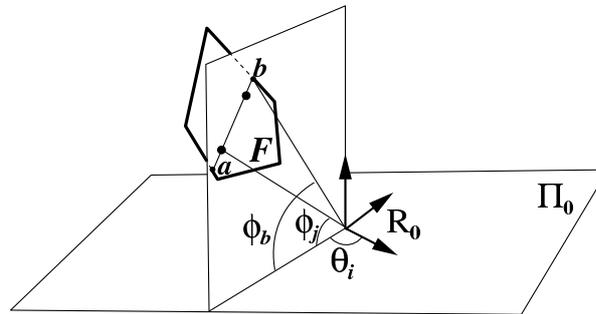


FIG. 2.13 – On détermine les échantillons dans l'intersection ab de la face F avec le quart de plan de coupe associé à l'angle θ_i .

En résumé, l'échantillonnage d'un objet polyédrique s'effectue en deux étapes :

1. l'attachement du polyèdre sur l'axe,
2. un parcours des faces du polyèdre pour trouver les échantillons qu'elles contiennent.

Remarque 2.3 Pour permettre à l'utilisateur d'employer des objets "ouverts" au niveau des calottes, nous avons adapté la procédure d'échantillonnage. Dans le cas d'objets ouverts, les échantillons obtenus ne remplissent pas entièrement les trois grilles de paramètres (c.f. figure 2.14). Or ce remplissage est nécessaire pour la mise en correspondance. De plus le bord de

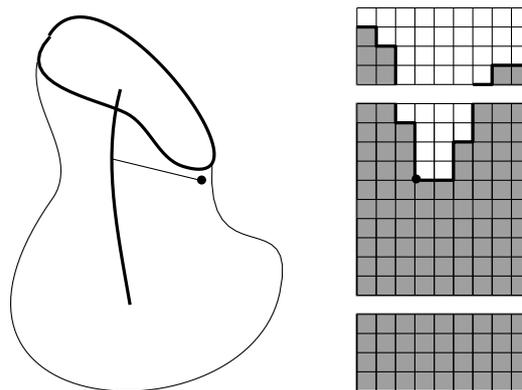


FIG. 2.14 – Un objet ouvert au niveau de la calotte supérieure. Les échantillons trouvés dans les cartes correspondent à la partie grise des cartes.

l'objet risque d'apparaître en dents de scie en raison de la répartition des échantillons. Pour

y remédier, nous déplaçons “verticalement” les échantillons du bord des cartes sur le bord de l’objet. Puis, nous finissons de remplir les tableaux des échantillons en dupliquant les valeurs du bord dans chaque colonne.

Nous pouvons ainsi échantillonner des objets homéomorphes à des sphères, des disques ou des cylindres. En fait le graphe correspondant aux objets échantillonnés est toujours cylindrique et nous ne réalisons pas d’opération topologique au cours des métamorphoses. Par contre les échantillons du bord supérieur du graphe d’une calotte fermée sont toujours géométriquement identifiés (c.f. figure 2.15) (un cylindre n’est jamais qu’une sphère privée de deux points).

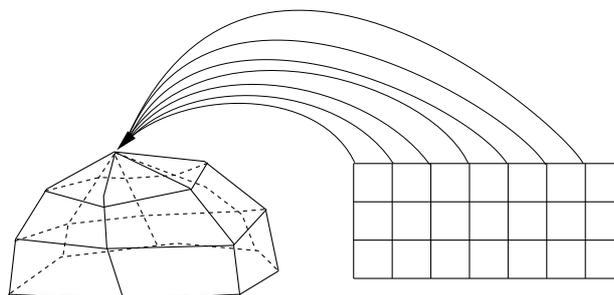


FIG. 2.15 – Les échantillons de paramètres $\phi_i = \frac{\pi}{2}$ sont identifiés.

Remarque 2.4 *Pour assurer le remplissage des grilles de paramètres, il est primordial de regarder les problèmes numériques qui sont inhérents aux algorithmes géométriques de ce type. Nous détaillons ci-dessous comment prendre en compte ces problèmes.*

Les deux opérations déterminantes pour l’échantillonnage sont :

1. intersecter une face avec un plan de coupe,
2. trouver le secteur angulaire, contenant les échantillons, associé à cette intersection.

Pour ne pas avoir de “trou” dans les grilles de paramètres, il faut être sûr que les secteurs angulaires associés à toutes les faces qui intersectent un plan de coupe Π_i recouvrent l’intervalle $[0, 2\pi[$. Nous utilisons pour cela une procédure qui effectue les calculs avec une précision ϵ et qui rend un secteur angulaire pour toute face qui intersecte le plan de coupe Π_i “à ϵ près”.

La procédure d’intersection d’une face avec un plan Π_i teste l’intersection de chaque arête de la face avec ce plan. Ce calcul est effectué avec la précision ϵ , de sorte qu’une arête qui intersecte effectivement Π_i soit détectée. Une arête ab est ainsi sélectionnée si elle vérifie :

$$I(ab, \Pi_i) < \epsilon$$

où I est une procédure qui calcule à ϵ près.

Lorsqu'une arête ab est sélectionnée, nous lui associons un angle qui correspond au paramètre θ_c de son intersection avec Π_i (c.f. figure 2.16). La précision de ce calcul n'est pas très

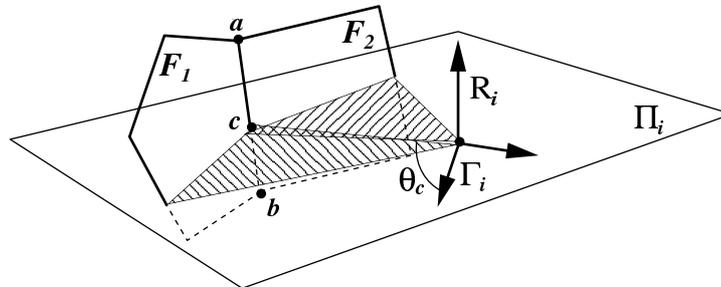


FIG. 2.16 – Les secteurs angulaires associés aux faces F_1 et F_2 sont figurés par des zones hachurées.

importante. Par contre, lors du traitement des faces F_1 et F_2 incidentes à l'arête ab , nous devons effectuer la *même* séquence de calculs sur l'arête ab (c.f. figure 2.16).

Le secteur angulaire associé à une face F est alors le plus grand secteur angulaire bordé par les angles θ_c associés à ces arêtes.

Il suffit ensuite d'intersecter le plan support de F avec les demi-droites d'échantillonnage incluses dans le secteur angulaire trouvé. De cette manière, tous les échantillons sont calculés au moins une fois. Certains échantillons peuvent être calculés plusieurs fois avec des faces différentes. Ces faces étant a priori proches l'une de l'autre, voire incidentes, les intersections des plans supports correspondants avec la demi-droite d'échantillonnage sont également proches (en supposant que les faces sont bien transversales aux demi-droites d'échantillonnage!).

2.4 Recalage

Si nous appliquons l'algorithme d'échantillonnage à un cube muni d'un axe droit comme sur la figure 2.17, l'objet résultant a un aspect très différent du cube d'origine: les côtés du cube n'apparaissent pas dans le modèle échantillonné.

Les éléments saillants d'un objet, ses arêtes vives et les points coniques, sont en effet de première importance dans l'apparence visuelle car ils démarquent des discontinuités dans l'intensité réfléchie.

Dans la méthode de J.R. Kent et al. [KPC91], ces éléments saillants apparaissent dans le sur-maillage reconstruit puisque ce sur-maillage contient exactement les maillages de chaque polyèdre à métamorphoser (voir l'état de l'art du chapitre 1). Par contre l'utilisation d'un

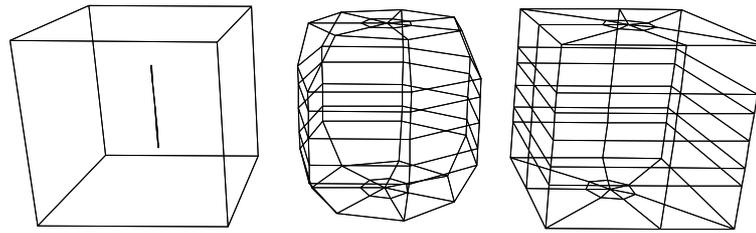


FIG. 2.17 – À gauche : un cube et son axe. Au milieu : le cube échantillonné avec la procédure précédente. À droite : le maillage recalé.

échantillonnage ne fait pas apparaître ces éléments dans le maillage rectangulaire reconstruit. Nous avons donc défini une étape de recalage du maillage échantillonné sur les éléments saillants des objets.

La procédure de recalage est constituée de deux étapes :

1. la sélection des éléments saillants,
2. la modification du maillage pour faire apparaître chaque élément saillant.

Sélection des éléments saillants

Dans notre implémentation, un critère angulaire entre les normales des deux faces incidentes à une arête permet de sélectionner les arêtes vives. Ce critère est simplement la comparaison de l'angle entre les normales avec un angle fixé par l'utilisateur (c.f. figure 2.18).

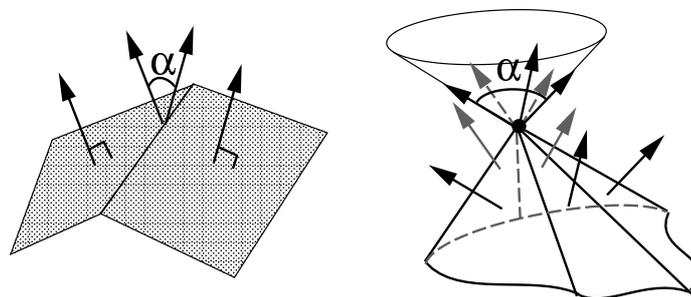


FIG. 2.18 – À gauche : l'angle α entre les normales des deux faces incidentes à une arête permet de déterminer si cette arête est vive. À droite : l'angle α maximal entre les normales des faces incidente à un sommet permet de le sélectionner comme sommet conique.

De même la sélection automatique des sommets coniques est effectuée à l'aide du "cône" des normales des faces incidentes à ce sommet.

Comme cette sélection automatique n'est généralement pas suffisante, nous permettons à l'utilisateur de sélectionner ou dé-sélectionner à l'écran des arêtes et des sommets du polyèdre. Ces éléments saillants sont rangés dans une liste ordonnée par l'ordre de sélection.

Recalage sur les éléments saillants

Nous cherchons à modifier le moins possible la topologie des maillages échantillonnés (trois grilles rectangulaires), afin de simplifier par la suite l'étape de mise en correspondance. Nous nous imposons donc de ne modifier le maillage que par les déplacements d'échantillons et par des ajouts de diagonales dans les mailles rectangulaires.

Le recalage est effectué de sorte qu'à la fin de cette étape :

- chaque arête vive est décomposée en une union disjointe d'arêtes du nouveau maillage,
- chaque point conique est un sommet du nouveau maillage.

Les déplacements des sommets échantillons sont locaux et bornés relativement à la paramétrisation. Plus précisément, nous découpons l'espace des paramètres en zones de déformation associées à chaque échantillon : la zone associée à l'échantillon (s_i, θ_j) est tout simplement le rectangle (c.f. figure 2.19) :

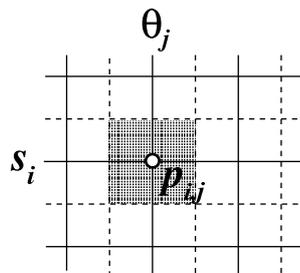


FIG. 2.19 – La zone de déplacement associée à un échantillon $p_{i,j}$ dans l'espace des paramètres.

$$\left[s_i - \frac{\Delta s}{2}, s_i + \frac{\Delta s}{2} \right[\times \left[\theta_j - \frac{\Delta \theta}{2}, \theta_j + \frac{\Delta \theta}{2} \right[$$

où Δs et $\Delta \theta$ sont les pas d'échantillonnage respectifs des paramètres s et θ .

Pour chaque élément saillant, nous localisons dans l'espace des paramètres les échantillons à déplacer que nous "projetons" sur l'élément saillant :

- pour chaque arête saillante \mathcal{A} nous effectuons un balayage similaire à celui utilisé dans l'étape d'échantillonnage. Dans la partie cylindrique, par exemple, \mathcal{A} est découpée par tous les plans de coupes *intermédiaires* (de paramètre $s_i + \frac{\Delta s}{2}$) et les paramètres θ_{a_i} des points d'intersection a_i sont utilisés pour déterminer les échantillons $p_{i,j}$ à déplacer.
 - Pour un échantillon $p_{i,j}$ de la partie cylindrique, nous calculons le point $q_{i,j}$ de \mathcal{A} le plus proche de la demi-droite d'échantillonnage correspondante. Si les paramètres de $q_{i,j}$ sont dans la zone de déformation associée à $p_{i,j}$, alors $p_{i,j}$ est déplacé sur ce point. Sinon, $p_{i,j}$ est déplacé sur le point déterminé par l'intersection de \mathcal{A} avec les frontières de la zone de déformation associée à $p_{i,j}$.
 - Dans le cas des calottes, nous utilisons une projection sphérique. Nous déterminons le point $q_{i,j}$ de \mathcal{A} , tel que sa projection sur la sphère unité, centrée à l'extrémité de l'axe, soit la plus proche de la projection de $p_{i,j}$ (c.f. figure 2.20). Si les paramètres de $q_{i,j}$ sont dans la zone de déformation associée à $p_{i,j}$, alors

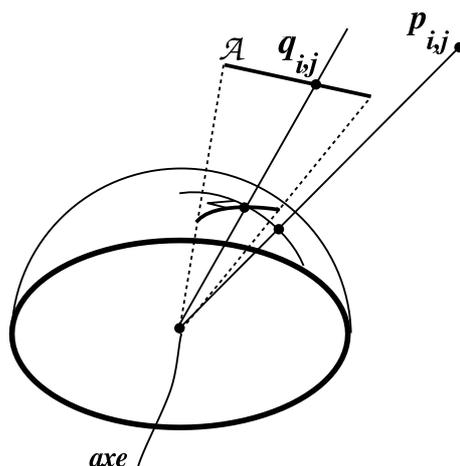


FIG. 2.20 – Le point $q_{i,j}$ est l'intersection de la droite support de \mathcal{A} et de la droite déterminée par les deux plans orthogonaux passant par l'extrémité de l'axe et contenant respectivement l'arête \mathcal{A} et le point $p_{i,j}$.

$p_{i,j}$ est déplacé sur ce point. Sinon, $p_{i,j}$ est déplacé sur le point déterminé par l'intersection de \mathcal{A} avec les frontières de la zone de déformation associée à $p_{i,j}$.

Après avoir déplacé les sommets échantillons sur l'arête vive \mathcal{A} , il se peut que le maillage déformé ne recouvre pas entièrement \mathcal{A} . C'est le cas dès que la séquence de sommets $p_{i,j}$ sélectionnée ne forme pas un chemin connexe dans les grilles d'échantillonnages. Dans ce cas, une diagonale doit être ajoutée pour chaque "trou" rencontré dans la

séquence de sommets $p_{i,j}$. Par exemple sur la figure 2.21, quatre diagonales ont été ajoutées, une pour chaque saut entre coupes, i.e., entre A et B, B et C, C et D et entre D et E.

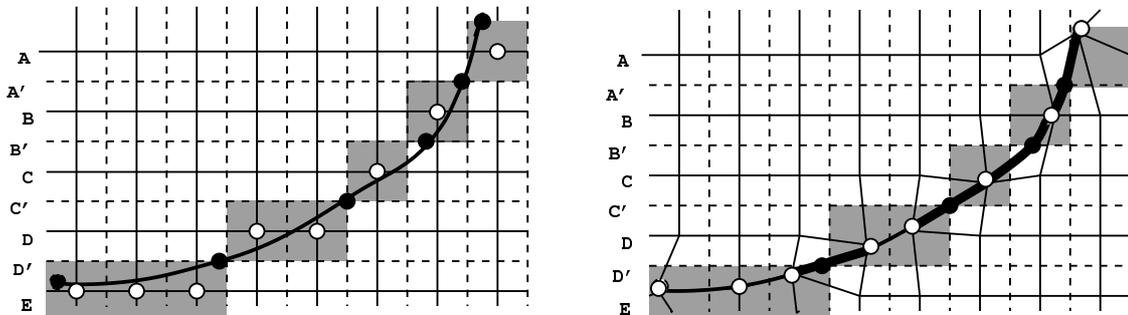


FIG. 2.21 – La courbe noire représente la projection d’une arête dans l’espace des paramètres. Les points blancs sont les échantillons à déplacer et leur zone de déplacement est représentée par des rectangles gris. Les droites horizontales en pointillés indiquent les plans de coupe intermédiaires, tandis que les points noirs sont les intersections de ces plans avec l’arête considérée (et les extrémités de l’arête). Sur la figure de droite on montre le maillage déformé et les diagonales ajoutées sont indiquées en trait épais.

- pour chaque point conique et chaque point extrémité d’arête vive, nous localisons ces paramètres dans une zone de déformation et déplaçons l’échantillon $p_{i,j}$ correspondant sur ce point.

Lorsque la finesse de l’échantillonnage n’est pas suffisante, plusieurs éléments saillants peuvent déformer successivement les mêmes sommets échantillons. Par conséquent, seul le recalage des derniers éléments saillants apparaîtra dans le maillage final s’il y a conflit. L’ordre de la liste des éléments saillants est donc de première importance et peut être modifié dans l’étape de sélection comme nous l’avons remarqué plus haut.

Notons qu’après l’étape de recalage, un sommet du maillage a au plus huit voisins (c.f figure 2.22). Nous ne pouvons donc pas recaler un maillage, aussi fin soit-il, sur un sommet conique incident à plus de huit arêtes vives. De tels sommets peuvent être détectés et nous pouvons modifier localement le maillage en conséquence.

En conclusion de l’état de l’art nous avons énuméré les avantages d’un échantillonnage par rapport à la construction d’un sur-maillage, à savoir l’adaptabilité à divers modes de représentation de surface, ou encore la possibilité d’interpoler une suite d’objets sans augmenter la complexité du maillage. En contrepartie, nous devons traiter séparément les éléments saillants. Remarquons tout de même qu’il est également possible de construire un

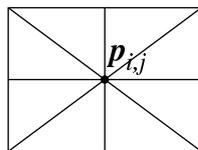


FIG. 2.22 – Un sommet $p_{i,j}$ du maillage recalé peut avoir huit voisins si des diagonales ont été ajoutées dans chacune des mailles incidentes.

sur-maillage dans un esprit semblable à la méthode de J.R. Kent et al. [KPC91, KCP92]. Pour cela, il faut être capable de dire si deux arêtes s'intersectent dans l'espace des paramètres et savoir projeter cette intersection sur chaque arête. Dans [KPC91], ce calcul est très simple et se réduit au calcul de l'intersection de deux plans contenant chacun une arête et le centre d'étoilement des objets. Dans notre cas, le calcul est moins simple mais peut être effectué par un balayage semblable à celui que nous avons utilisé pour le recalage : nous intersectons les rectangles de paramètres définis par les deux extrémités de chaque arête et nous réduisons progressivement l'intervalle de coupe associé si cette intersection est non vide.

2.5 Mise en correspondance et interpolation

2.5.1 Mise en correspondance

Après recalage des deux objets à métamorphoser sur leurs éléments saillants, leurs maillages échantillonnés n'ont plus nécessairement la même topologie. En effet, certaines mailles, initialement rectangulaires, ont pu être coupées en deux par une diagonale.

Dans le tableau $p_{i,j}$ qui représente le maillage d'un objet, nous avons réservé un champ qui indique si une diagonale a été introduite dans la maille rectangulaire $(p_{i,j}, p_{i+1,j}, p_{i+1,j+1}, p_{i,j+1})$. Il y a deux types de diagonales. La diagonale de type M relie $p_{i,j}$ et $p_{i+1,j+1}$, tandis que celle de type D relie $p_{i+1,j}$ et $p_{i,j+1}$.

Pour construire un nouveau sur-maillage qui contienne les diagonales introduites dans les deux objets, il suffit de parcourir les tableaux $p_{i,j}^0$ et $p_{i,j}^1$ associés à chaque objet :

- si les deux échantillons $p_{i,j}^0$ et $p_{i,j}^1$ pointent respectivement sur des diagonales de type M et D , nous coupons la maille (i, j) du nouveau maillage en quatre triangles (c.f. figure 2.23),
- si un seul type de diagonale apparaît, nous coupons en deux la maille (i, j) du nouveau maillage avec la même diagonale.

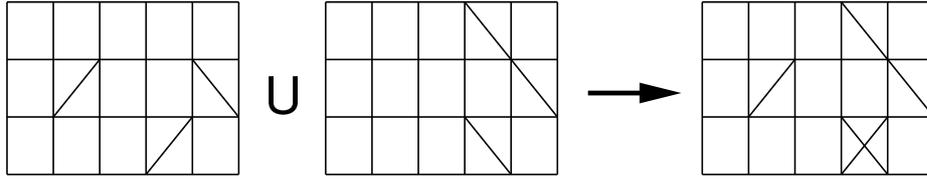


FIG. 2.23 – Le surmaillage est obtenu par superposition des grilles de chaque objet recalé.

À ce niveau, il est possible de trianguler le reste des mailles qui ne contiennent pas de diagonale pour assurer la planarité des faces de l'approximation polyédrique.

2.5.2 Interpolation

Lors de l'étape d'échantillonnage, les coordonnées des échantillons sont calculées dans le repère local associé à la coupe considérée. Par exemple, si l'échantillon $p_{i,j}$ de la partie cylindrique a pour coordonnées locales $(x_{i,j}, y_{i,j}, z_{i,j})$, ses coordonnées dans le repère du "monde" sont :

$$\Gamma_i + R_i \begin{pmatrix} x_{i,j} \\ y_{i,j} \\ z_{i,j} \end{pmatrix}$$

Lorsque l'échantillon $p_{i,j}$ n'est pas déplacé par l'étape de recalage, nous avons d'ailleurs :

$$x_{i,j} = 0 \quad y_{i,j} = r_{i,j} \cos \theta_j \quad z_{i,j} = r_{i,j} \sin \theta_j$$

La métamorphose de deux objets s'obtient en interpolant simultanément les systèmes de coordonnées (Γ_i, R_i) , i.e. les axes munis de leur repère, et les coordonnées locales des points.

- Les axes et leurs repères sont interpolés à l'aide du procédé décrit dans la première partie de ce manuscrit.
- Les coordonnées locales sont interpolées à l'aide d'une fonction monotone $\alpha_i(t)$ définie le long de l'axe :

$$\begin{pmatrix} x_{i,j}^t \\ y_{i,j}^t \\ z_{i,j}^t \end{pmatrix} = (1 - \alpha_i(t)) \begin{pmatrix} x_{i,j}^0 \\ y_{i,j}^0 \\ z_{i,j}^0 \end{pmatrix} + \alpha_i(t) \begin{pmatrix} x_{i,j}^1 \\ y_{i,j}^1 \\ z_{i,j}^1 \end{pmatrix} \quad (2.2)$$

Si les axes peuvent être munis du repère de Frenet, nous pouvons assurer une paramétrisation régulière dans la partie cylindrique en choisissant :

$$\alpha_i(t) = \frac{1}{(1 - \frac{v_{0,i}}{v_{1,i}}) + \frac{1}{t} \frac{v_{0,i}}{v_{1,i}}} = \frac{t}{t + (1 - t) \frac{v_{0,i}}{v_{1,i}}}$$

En effet, dans ce cas, nous n'aurons pas d'auto-intersection locale car la courbure interpolée vérifie :

$$\kappa_{t,i} = (1 - \alpha_i(t))\kappa_{0,i} + \alpha_i(t)\kappa_{1,i}$$

comme nous l'avons vu dans la première partie de ce manuscrit.

Dans la pratique nous utilisons, pour $\alpha_i(t)$, le champ scalaire défini sur les axes (c.f. partie II, §2.3.2: Contrôle de la vitesse de transformation). Sur les images extraites du film METADATA [PLVB94] (c.f. figure ??) nous voyons par exemple un cube se transformer en tête de manière non uniforme le long d'un axe rectiligne interne à l'objet.

Raccord entre les calottes et la partie cylindrique

Nous allons voir qu'il n'est généralement pas possible d'obtenir, au cours de l'interpolation, un raccord G^1 -continu entre les calottes et la partie cylindrique, même si les objets clefs sont lisses. Pour cela, nous pouvons supposer que les objets sont attachés à un même axe vertical en utilisant le redressement par le système de coordonnées étudié dans la section 2.2. Comme ce redressement laisse invariant chaque plan de coupe, on peut toujours affirmer que l'interpolation est radiale par rapport à l'axe.

Chaque objet admet une paramétrisation du type :

$$(s, \theta) \mapsto s\mathbf{e}_1 + r(s, \theta)\mathbf{e}_\theta$$

dans la partie cylindrique (c.f. figure 2.24 pour les notations) et une paramétrisation du type :

$$(\theta, \phi) \mapsto \mathbf{e}_1 + \rho(\theta, \phi)\mathbf{e}_{\theta, \phi}$$

dans la calotte supérieure (nous raisonnons sur le raccord avec la calotte supérieure). L'hypothèse de transversalité des objets par rapport aux demi-droites d'échantillonnage permet d'affirmer que les fonctions r et ρ sont différentiables. On vérifie assez simplement que la condition de continuité de la normale au niveau du raccord ($s = 1$ et $\phi = 0$) s'exprime par l'équation :

$$\frac{\partial \rho}{\partial \phi} = r \frac{\partial r}{\partial s} \quad (2.3)$$

L'interpolation définie par l'équation 2.2 nous fournit deux fonctions r_t et ρ_t :

$$r_t = (1 - \alpha(t))r_0 + \alpha(t)r_1$$

$$\rho_t = (1 - \alpha(t))\rho_0 + \alpha(t)\rho_1$$

où (r_0, ρ_0) et (r_1, ρ_1) sont les rayons associés aux objets clefs. L'équation de raccord 2.3 à l'instant t devient, en tenant compte du fait qu'elle est vérifiée aux instants 0 et 1 :

$$\frac{1}{2} \alpha(t) (1 - \alpha(t)) \frac{\partial}{\partial s} (r_0 - r_1)^2 = 0$$

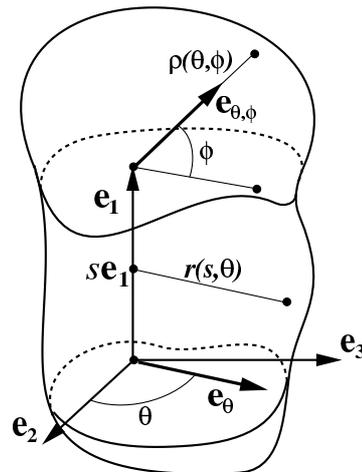


FIG. 2.24 – Paramétrisation de la partie cylindrique et des calottes.

Ceci montre que les objets interpolés admettent un raccord G^1 si et seulement si pour tout couple de points en correspondance sur les bords des objets, les droites verticales tangentes en ces points sont parallèles (c.f. figure 2.25). Notons que ce problème est identique à celui

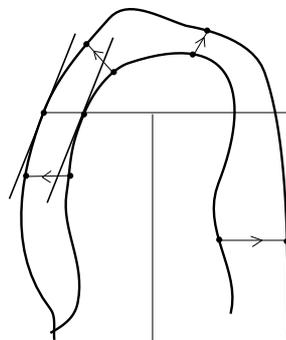


FIG. 2.25 – La figure représente une coupe des objets par un plan vertical passant par l'axe. Les points en correspondance sur le bord doivent avoir des tangentes parallèles pour obtenir une interpolation régulière.

rencontré dans le plan pour interpoler des courbes partiellement exprimées en coordonnées cartésiennes et en coordonnées polaires.

Dans la pratique, le défaut de “lissité” est faible lorsque l'angle entre les droites précédentes n'est pas trop important comme on peut s'en convaincre sur les figures 2.26 et 2.27.

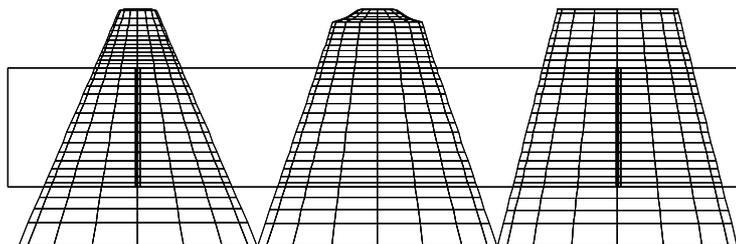


FIG. 2.26 – Transformation entre deux cônes tronqués. Les axes sont identiques et visualisés en traits épais. Les objets sont représentés aux instants 0, 1/2 et 1. Le grand rectangle indique le niveau de raccord entre la partie cylindrique et les calottes.

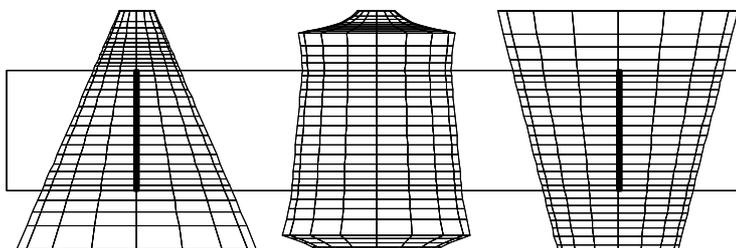


FIG. 2.27 – Transformation entre deux cônes tronqués. Les angles entre les plans tangents aux deux cônes sont plus importants que sur la figure 2.26.

2.6 Exemples de métamorphoses

Nous présentons ici quelques exemples de métamorphoses. L'ensemble de nos implémentations s'insère dans le modèleur ACTION3D, un système de modélisation et d'animation développé conjointement par l'INRIA et la société SOGITEC. Les métamorphoses que nous présentons ont toutes été réalisées en temps interactif sur des machines SGI INDIGO 2.

- Les figures 2.28, 2.29 et 2.30 illustrent le rôle de l'axe dans le contrôle des déformations. Deux axes ont été utilisés : un segment vertical et une courbe oblique. Le niveau de discrétisation est volontairement faible pour mieux visualiser les maillages : les axes comportent 7 sommets, les angles θ et ϕ sont respectivement discrétisés avec 16 et 5 échantillons. Sur les figures en fils de fer 2.28 et 2.29, les axes sont représentés en trait épais pour les formes clefs (initiales et finales).

Sur la figure 2.28, le même axe, un segment vertical, est utilisé pour échantillonner les formes clefs. Dans la transformation du bas de la figure 2.28, l'axe a été translaté à l'intérieur du cube. Ce mode de contrôle est comparable au contrôle des transformations

de J.R. Kent et al. [KPC91] par translation du centre d'étoilement à l'intérieur des objets.

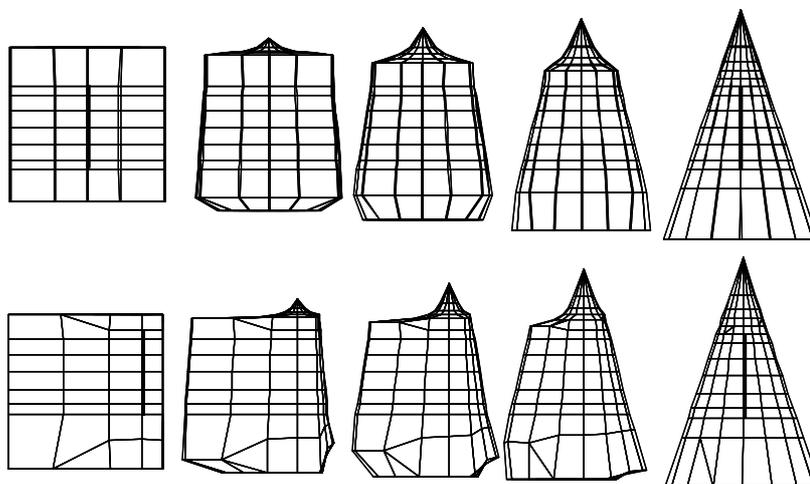


FIG. 2.28 – Le même axe est utilisé pour échantillonner les deux objets clefs.

Lorsque les axes associés aux formes clefs sont distincts, comme sur la figure 2.29, l'interpolation de l'axe influe notablement sur la transformation. Dans la figure 2.29, nous avons échangé l'axe initial et final entre les transformations du haut et du bas de l'image. Notons que le faible taux d'échantillonnage permet tout de même d'obtenir des images intéressantes (c.f. figure 2.30).

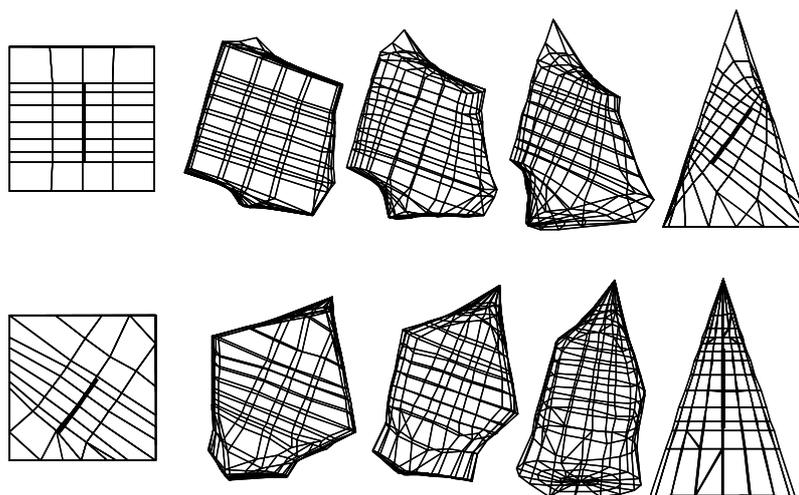


FIG. 2.29 – L'axe initial et l'axe final sont échangés entre la transformation de la rangée du haut et celle de la rangée du bas.

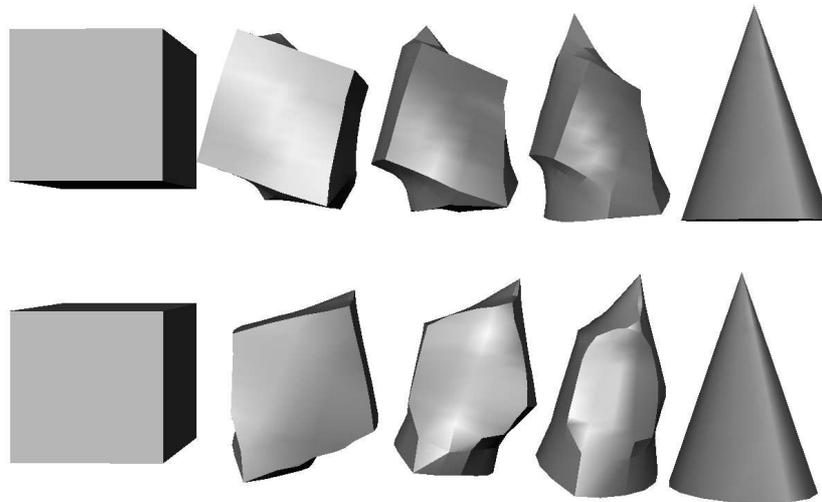


FIG. 2.30 – Un rendu simple de la figure 2.29.

- Sur la figure 2.31, nous montrons trois transformations, en colonne, entre un cube, un poisson et un tube courbé. Nous pouvons noter que les objets interpolés ne s'auto-intersectent pas. Les axes sous-jacents sont rectilignes pour le cube et le poisson.

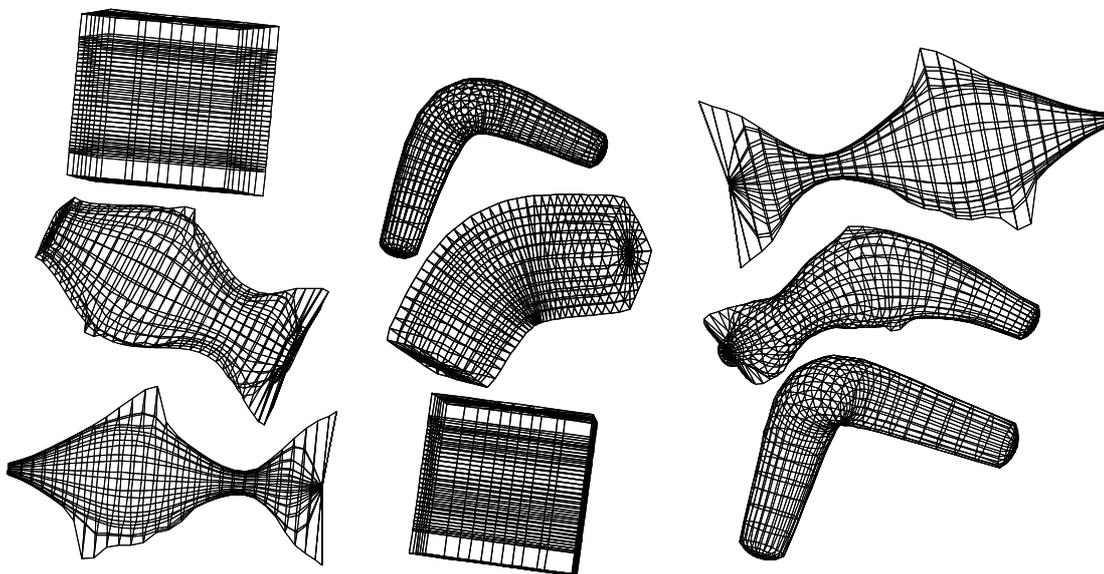


FIG. 2.31 – Les formes de la rangée du milieu sont interpolées entre les formes du haut et du bas.

- Les figures 2.32 et 2.33 illustrent la possibilité d'avoir des objets ouverts. Nous pouvons

de plus noter la position oblique de l'axe par rapport au bord de la demi-sphère utilisée.



FIG. 2.32 – Transformation d'une calotte sphérique en sphère.

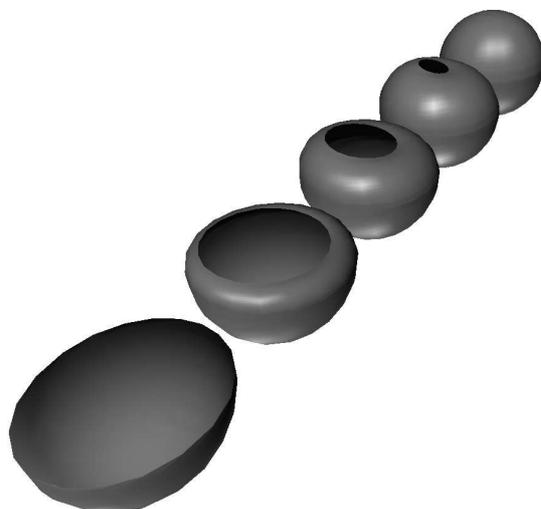


FIG. 2.33 – Un rendu de la figure 2.32.

- La figure 2.34 montre la transformation d'une faucille en poisson. Sur ces images la transformation apparaît clairement lisse.
- Nous terminons sur des exemples plus compliqués tirés du film METADATA [PLVB94]. Le maillage échantillonné associé à chaque objet comporte 7200 facettes : l'axe est approximé par 50 points, tandis que chaque section est constituée de 80 échantillons. Les parties hémisphériques sont composées de 1600 facettes (80×20). Le cygne apparaissant dans ces images est échantillonné avec un axe qui suit sa courbure naturelle. Les autres formes sont échantillonnées avec des axes approximativement rectilignes. Les quatre transformations illustrent des effets visuels particuliers tels que :
 - l'importance des éléments saillants dans la transformation avec le piano,

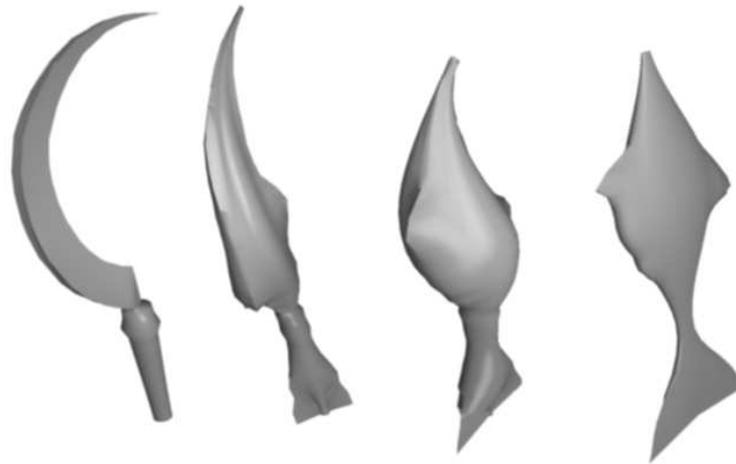


FIG. 2.34 – Transformation d'une faucille en poisson.

- l'interpolation des axes dans la transformation avec le cygne,
- l'intérêt du positionnement des axes dans la transformation des têtes : les nez et les yeux des deux têtes se correspondent,
- l'introduction d'un contrôle de la vitesse de transformation le long de l'axe : ici le cube est graduellement transformé en tête, en partant du haut vers le bas. Ce contrôle est comparable à celui utilisé dans les transformations d'images pixels par S-Y Lee et al. [LCHS94].

FIG. 2.35 – *Deux métamorphoses extraites du film Metadata.*FIG. 2.36 – *Deux métamorphoses extraites du film Metadata.*

2.7 Conclusion

Dans cette dernière partie, nous avons défini un procédé de métamorphose pour des objets de forme tubulaire. Ce procédé s'appuie sur quatre étapes distinctes :

1. l'attachement des objets clefs sur leur axe respectif,
2. l'échantillonnage de ces objets à partir de leur axe et de l'attachement de la première étape,
3. le recalage éventuel des maillages obtenus sur les éléments saillants des objets,
4. l'interpolation proprement dite des maillages ainsi construits.

Nous avons étudié avec précision chaque étape, en faisant ressortir les problèmes numériques ou conceptuels liés à chacune d'elles. En particulier, l'étude de l'attachement d'un polyèdre à une courbe polygonale fournit un algorithme d'attachement qui nous semble original et efficace.

Notre méthode offre l'avantage d'un contrôle important sur la transformation. L'utilisateur peut varier la forme et la position des axes internes aux objets, dans la mesure où la condition d'étoilement reste vérifiée. Ces variations lui offrent une certaine liberté d'expression comme le montrent les exemples de la section 2.6. Une fois les axes fixés dans chaque objet, il peut également jouer sur la transformation par l'intermédiaire de la paramétrisation des axes, modifiant ainsi les appariements entre coupes, puis en modifiant l'interpolation des axes comme nous l'avons montré dans la première partie de ce mémoire.

Cette technique de métamorphose a été employée par A. Paouri, une artiste infographiste, pour réaliser une séquence animée [PLVB94]. Les effets produits dans cette séquence n'auraient pas pu être obtenus avec les techniques de métamorphoses offertes jusqu'à présent.

Conclusion

Nous avons proposé une nouvelle méthode pour réaliser une métamorphose entre deux objets surfaciques de forme tubulaire. Cette restriction à une forme tubulaire nous a permis de considérer les objets comme des immersions explicites du cylindre $S^1 \times D^1$. Nous avons ainsi ramené le problème de l'interpolation d'objets à la recherche d'une homotopie entre deux immersions du cylindre. Plus précisément, nous avons exprimé ces immersions sous forme de cylindres généralisés, réduisant à nouveau cette recherche essentiellement à celle d'une interpolation de courbes.

Dans un premier temps, nous avons proposé un procédé d'interpolation de courbes pour l'animation sous un angle original qui n'avait pas été envisagé dans notre discipline. Partant d'une étude précise et mathématique de l'interpolation nous avons défini un algorithme permettant de contrôler par des paramètres simples les caractéristiques géométriques de l'interpolation.

Par ailleurs, le fait que nos objets ne soient pas donnés au départ sous forme de cylindre généralisé fait l'objet des deuxième et troisième parties puisque nous devons justement construire une courbe squelette dans chaque forme et attacher celle-ci à cette courbe par l'intermédiaire d'un système de coordonnées approprié.

En particulier, la recherche d'une courbe squelette, ou axe, dans un objet polyédrique nous a mené à développer dans la deuxième partie une analogie combinatoire simple avec la théorie des fonctions de Morse, à travers la notion de coloriage d'un graphe à valeurs dans un ensemble ordonné. Nous avons ainsi proposé une formule d'Euler et une décomposition cylindrique des polyèdres. Nous avons alors étudié précisément le cas où le graphe est colorié par une distance géodésique discrète, et nous en avons déduit, par moyennage du découpage en bande associé, une définition d'axe originale et plus appropriée que celle d'axe médian.

Dans la troisième partie, nous avons présenté une technique de métamorphose qui repose sur l'interpolation de courbes étudiée dans la première partie. L'axe, relativement auquel l'objet doit être étoilé, peut être défini interactivement à partir de l'axe proposé dans la seconde partie. Les objets clefs sont échantillonnés par l'intermédiaire d'une paramétrisation liée à leur axe respectif. Nous détaillons ces étapes de paramétrisation et d'échantillonnage

et proposons pour ce faire un algorithme d'attachement (i.e. de projection) d'un polyèdre sur une courbe polygonale.

En partant du principe que les discontinuités d'une forme sont essentielles dans son aspect, nous établissons finalement une étape de recalage d'une forme échantillonnée sur ces éléments saillants.

D'une façon générale, notre souci d'exprimer les problèmes de manière intrinsèque nous a permis d'obtenir des algorithmes simples et robustes aux paramètres clairement identifiés. Nous avons pu ainsi définir un outil interactif de métamorphose caractérisé par

- sa simplicité d'utilisation,
- la modularité de son contrôle par l'intermédiaire de l'axe et de son interpolation,
- l'aspect intuitif de ses paramètres.

Cet outil a de plus été validé par la réalisation d'une séquence animée [PLVB94].

Notons que malgré l'intérêt intrinsèque de l'axe proposé dans la deuxième partie, il faudrait encore réfléchir sur son adaptation à la paramétrisation d'une forme. Hormis une étape de recentrage de cet axe, nous pouvons imaginer de conserver pour fibres les cercles géodésiques qui ont servi à sa construction au lieu des sections planes et orthogonales à l'axe. Il faudrait alors être au moins capable d'interpoler deux fibres, c'est-à-dire deux courbes gauches compactes (i.e. fermées). Cette dernière condition rend malheureusement la tâche beaucoup plus ardue que celle traitée dans la première partie. Le seul cas de courbes planes fermées, a priori non étoilées, pose des difficultés comme le montrent les méthodes de métamorphose de contours décrites dans l'état de l'art de la troisième partie.

Par ailleurs nous avons montré dans la troisième partie que le raccord entre la partie cylindrique d'un objet et ses calottes n'étaient généralement pas G^1 au cours de l'interpolation. D'une manière générale les calottes et les embranchements (du type cactus) posent des problèmes qui entraînent le recours à des modèles spécifiques tels que ceux proposés par J. Séqueira, R. Ebel et F. Schmitt [SES93, Ebe92] ou par M.-P. Gascuel [Gas90]. En effet, cette caractérisation des formes par des calottes ou des embranchements est liée à l'utilisation d'une projection définie au moyen d'une fonction de Morse sur les objets. Les singularités de ces projections font que les fibres associées, i.e. les ensembles qui se projettent sur un même point, peuvent changer de topologies, de sorte qu'il devient difficile de mettre en relation les fibres de deux objets.

Ainsi la simplification apparente fournie par l'utilisation de projections peut-elle s'avérer source de difficultés dans le cadre d'une interpolation entre deux formes quelconques. Elle est néanmoins nécessaire si l'on veut donner un sens au mot forme, c'est à dire être capable

de différencier les divers plongements d'une variété. Notre perception des formes est en effet intimement liée à l'utilisation de projections comme en témoigne notre système visuel. Plus exactement elle est liée à une famille de projections comme, par exemple, la famille des projections orthogonales d'une surface [Ker81]. Cette famille de projections caractérise les divers aspects d'une forme, et la théorie des singularités nous apprend que ces aspects (i.e. les paramètres de la famille de projections) se stratifient génériquement⁵ en complexes cellulaires, dont les cellules définissent des aspects stables (i.e. constituent des classes d'isotopie). Nous pouvons dès lors esquisser un programme de recherche pour l'élaboration d'un outil de métamorphoses plus général :

1. Il faut avant tout choisir une famille de projections qui détermine la forme. On peut par exemple exiger que cette famille distingue un cactus à trois branches d'un cactus à cinq branches.
2. Puis nous devons étudier la structure des diverses décompositions cellulaires associées à cette famille de projections. Pour être exploitable, une famille de projections devra fournir des décompositions suffisamment simples pour être "traitables" dans les cas pratiques.
3. Par la suite nous devons fixer une famille raisonnable de complexes cellulaires qui fournisse un ensemble de formes modèles assez vaste pour être utile à un animateur. Ces modèles devront être constructibles canoniquement ou à défaut être mémorisés une fois pour toute.
4. Il faudra de plus définir des déformations entre modèles, ce qui d'une certaine manière revient à déformer les graphes d'aspects associés.
5. Enfin, nous devons exprimer comment passer entre deux formes qui ont les mêmes aspects afin de ramener un objet que l'on cherche effectivement à métamorphoser sur un modèle de même aspect.

Il nous semble que c'est à ce prix uniquement que nous pourrions définir un outil assez puissant pour traiter un ensemble de formes suffisamment générique.

5. i.e. pour un ouvert dense de l'espace des plongements muni d'une topologie adéquate.

Annexe A

Linéarisation d'un champ invariant

Proposition A.1 *Soit X un groupe de Lie admettant une représentation linéaire fidèle dans $GL(n, \mathbb{R})$, notée ρ et \mathbf{V} un champ sur X dépendant du temps et invariant à tout instant (noté s), c'est à dire de la forme¹ :*

$$\mathbf{V}(x, s) = \theta_x^{-1} \cdot \mathbf{v}(s)$$

où $\mathbf{v}(s)$ est une fonction de classe C^1 à valeurs dans $T_e X$, l'espace tangent en l'unité e du groupe. Alors \mathbf{V} est transformé par ρ en un champ image $\tilde{\mathbf{V}}$ sur $\rho(X)$ qui se prolonge sur l'espace $\mathcal{M}(n, \mathbb{R})$ des matrices de taille $n \times n$ en un champ linéaire et donc intégrable.

Démonstration : Par définition d'un champ image, on a pour tout \tilde{x} de $\rho(X)$

$$\tilde{\mathbf{V}}(\tilde{x}, s) = \rho^* \mathbf{V}(\tilde{x}, s) = T_x \rho \cdot \mathbf{V}(x, s)$$

où $x = \rho^{-1}(\tilde{x})$ (ρ est injective). Notons que l'on a

$$\theta_x^{-1} = (T_x \tau_x)^{-1} = T_{\tau_x(x)}(\tau_x^{-1}) = T_e \tau_{x^{-1}}$$

D'où :

$$\begin{aligned} \tilde{\mathbf{V}}(\tilde{x}, s) &= T_x \rho \cdot T_e \tau_{x^{-1}} \cdot \mathbf{v}(s) \\ &= T_e (\rho \circ \tau_{x^{-1}}) \cdot \mathbf{v}(s) \end{aligned}$$

Or pour tout y de X

$$\rho \circ \tau_{x^{-1}}(y) = \rho(xy) = \rho(x) \rho(y) = \tilde{\tau}_{\tilde{x}^{-1}} \circ \rho(y)$$

où $\tilde{x} = \rho(x)$ et $\tilde{\tau}_{\tilde{x}}$ désigne les translations de $GL(n, \mathbb{R})$. D'où

$$\begin{aligned} \tilde{\mathbf{V}}(\tilde{x}, s) &= T_e (\tilde{\tau}_{\tilde{x}^{-1}} \circ \rho) \cdot \mathbf{v}(s) \\ &= T_{\tilde{e}} \tilde{\tau}_{\tilde{x}^{-1}} \circ T_e \rho \cdot \mathbf{v}(s) \end{aligned}$$

1. θ_x est l'isomorphisme de transport défini dans la section 1.4.1 de la première partie

où \tilde{e} est l'unité de $GL(n, \mathbb{R})$.

En notant

$$\tilde{\mathbf{v}} = T_e \rho \cdot \mathbf{v}$$

on en déduit

$$\tilde{\mathbf{V}}(\tilde{x}, s) = T_{\tilde{e}} \tilde{\tau}_{\tilde{x}^{-1}} \cdot \tilde{\mathbf{v}}(s) = \tilde{\theta}_{\tilde{x}}^{-1} \cdot \tilde{\mathbf{v}}(s)$$

Remarquons alors que $\tilde{\tau}_{\tilde{x}^{-1}}$ peut être prolongé sur $\mathcal{M}(n, \mathbb{R})$ ($\supset GL(n, \mathbb{R})$) en un opérateur linéaire. Par conséquent sa différentielle s'identifie à lui-même et $\tilde{\mathbf{V}}$ peut s'écrire

$$\tilde{\mathbf{V}}(\tilde{x}, s) = \tilde{x} \cdot \tilde{\mathbf{v}}(s)$$

qui est bien linéaire en \tilde{x} \square

Proposition A.2 *Le champ $\mathbf{V}(x, s)$ est intégrable sur X .*

Soit x une condition initiale sur X . D'après la proposition précédente on obtient une solution $\tilde{\gamma}$ de $\tilde{\mathbf{V}}$ de condition initiale $\tilde{x} = \rho(x)$. Il suffit alors de montrer que $\tilde{\gamma}$ reste sur $\rho(X)$ et que $\rho^{-1}(\tilde{\gamma})$ est une solution globale sur X de condition initiale x pour obtenir la proposition cherchée.

Or \mathbf{V} est localement intégrable sur X car \mathbf{V} est de classe C^1 au moins et X est localement compact. De plus les images par ρ de ces solutions locales sont des solutions locales de $\tilde{\mathbf{V}}$. On en déduit que toute solution locale γ de condition initiale $\gamma(0) = x$ coïncide localement avec $\rho^{-1}(\tilde{\gamma})$. En considérant alors le plus grand intervalle $[s_-, s_+]$ où ceci est vrai, on en déduit aisément que $\rho^{-1}(\tilde{\gamma})$ est globalement la solution cherchée \square

Annexe B

Γ_i^n converge uniformément vers Γ

On suppose que Γ_i^n est obtenue progressivement par l'équation :

$$\Gamma_{i+1}^n = \Gamma_i^n + \frac{1}{n} R_i^n \Delta_i^n$$

où R_i^n et Δ_i^n convergent respectivement vers un repère mobile R et un vecteur Δ , et $\Gamma_0^n = \Gamma_0$ est un point fixe.

On pose

$$\gamma(s) = \Gamma_0 + \int_0^s R(u) \Delta(u) du$$

Alors Γ_i^n converge vers γ , c'est-à-dire :

$$\lim_{n \rightarrow \infty} \sup_{s \in [0,1]} \|\Gamma_{[sn]}^n - \gamma(s)\| = 0.$$

Démonstration : Pour simplifier les notations, on pose

$$\delta_i^n = R_i^n \Delta_i^n \text{ et } \delta(u) = R(u) \Delta(u)$$

D'où

$$\|\Gamma_{[sn]}^n - \gamma(s)\| = \|\Gamma_0^n + \frac{1}{n} \sum_{k=0}^{[sn]-1} \delta_k^n - \Gamma(0) - \int_0^s \delta(u) du\|$$

Transformons la somme en intégrale

$$\frac{1}{n} \sum_{k=0}^{[sn]-1} \delta_k^n = \sum_{k=0}^{[sn]-1} \int_{\frac{k}{n}}^{\frac{k+1}{n}} \delta_{[un]}^n du = \int_0^{\frac{[sn]}{n}} \delta_{[un]}^n du$$

On en déduit, compte tenu du fait que $\Gamma_0^n = \Gamma(0)$

$$\|\Gamma_{t,[sn]}^n - \gamma(s)\| = \left\| \int_0^{\frac{[sn]}{n}} \delta_{[un]}^n du - \int_0^s \delta(u) du \right\|$$

$$\begin{aligned} &\leq \int_0^{\lfloor \frac{sn}{n} \rfloor} \|\delta_{[un]}^n - \delta(u)\| du + \int_{\lfloor \frac{sn}{n} \rfloor}^s \|\delta(u)\| du \\ &\leq \sup_{u \in [0,1]} \|\delta_{[un]}^n - \delta(u)\| + \frac{1}{n} \sup_{s \in [0,1]} \|\delta(s)\| \end{aligned}$$

On conclut en utilisant le fait que δ_i^n converge uniformément vers δ , et que δ est borné sur le compact $[0,1]$ \square

Annexe C

Solution numérique de $R' = R\tilde{\Omega}$

Dans ce qui suit on notera SO_3 , au lieu de $SO(3, \mathbb{R})$, l'espace des rotations de \mathbb{R}^3 .

Pour résoudre le système différentiel :

$$\begin{cases} R'(s) = R(s)\tilde{\Omega}(s) \\ R(0) = R_0 \end{cases} \quad (\text{C.1})$$

nous pouvons considérer R comme une matrice de l'espace vectoriel $\mathcal{M}(3, \mathbb{R})$ et utiliser la méthode d'Euler classique, avec un pas de résolution $\frac{1}{n}$, qui permet d'obtenir progressivement une approximation R_i de $R(\frac{i}{n})$ par l'équation :

$$R_{i+1} = R_i + \frac{1}{n} R_i \tilde{\Omega}\left(\frac{i}{n}\right) \quad (\text{C.2})$$

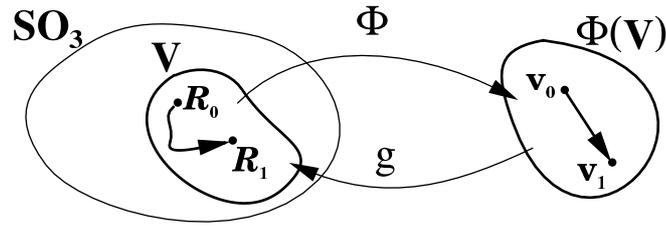
Cependant, même si R_i est une rotation, R_{i+1} n'est pas nécessairement une rotation car l'équation est considérée dans $\mathcal{M}(3, \mathbb{R})$ et non dans SO_3 . Ceci peut ou non être problématique, mais nous préférons résoudre directement le système différentiel C.1 dans SO_3 puisque $R\tilde{\Omega}$ peut être considéré comme un champ (invariant à tout "instant") sur SO_3 ¹. Pour cela nous allons considérer une carte locale de SO_3 afin de "ramener" le système C.1 dans un ouvert de \mathbb{R}^3 où nous pourrions utiliser la méthode d'Euler.

En effet, si (V, ϕ) est une carte de SO_3 autour d'une rotation R_0 , et $g = \phi^{-1}$ est la paramétrisation correspondante (voir figure C.1), le système C.1 se transforme via la paramétrisation en

$$\begin{cases} \mathbf{v}' = (dg)^{-1} \cdot g(\mathbf{v}) \tilde{\Omega} \\ \mathbf{v}_0 = \phi(R_0) \end{cases} \quad (\text{C.3})$$

Un moyen canonique de paramétrer SO_3 est d'utiliser l'application exponentielle parfois

1. L'isomorphisme de transport θ_R introduit dans la section 1.4.1 s'identifie en effet dans le cas de SO_3 à l'application $\tilde{\Omega} \mapsto R\tilde{\Omega}$.

FIG. C.1 – Une carte autour d'une rotation R_0 .

appelée application normale canonique. Notons que l'exponentielle peut être définie, dans le cas de SO_3 , aussi bien par la structure de groupe de Lie de SO_3 ² que par la structure riemannienne héritée de l'espace euclidien $\mathcal{M}(3, \mathbb{R})$ ³. Pour montrer la coïncidence des deux définitions, il suffit en effet de remarquer que SO_3 agit sur $\mathcal{M}(3, \mathbb{R})$ comme groupe d'isométrie par l'action

$$M \mapsto RM.$$

Autour de l'identité l'exponentielle prend la forme suivante :

$$\begin{aligned} (\mathbb{R}^3, 0) &\rightarrow (SO_3, Id) \\ \mathbf{v} &\mapsto \exp(\tilde{\mathbf{v}}) = Id + \tilde{\mathbf{v}} + \frac{\tilde{\mathbf{v}}^2}{2!} + \dots + \frac{\tilde{\mathbf{v}}^n}{n!} + \dots = Id + \frac{\sin \|\mathbf{v}\|}{\|\mathbf{v}\|} \tilde{\mathbf{v}} + \frac{1 - \cos \|\mathbf{v}\|}{\|\mathbf{v}\|^2} \tilde{\mathbf{v}}^2 \end{aligned}$$

Cette paramétrisation peut être translatée autour d'une rotation R quelconque :

$$\mathbf{v} \mapsto R \exp(\tilde{\mathbf{v}})$$

La différentielle de l'exponentielle s'obtient soit de manière algébrique comme dans [Fau89] soit plus simplement en différenciant le membre de droite dans l'expression de l'exponentielle ci-dessus (c'est peut-être moins joli mais incomparablement plus court). On a alors ($T_{Id}SO_3$ étant identifié à \mathbb{R}^3) :

$$d \exp(\tilde{\mathbf{v}}) \cdot \mathbf{h} = R \tilde{\mathbf{b}}(\mathbf{v}, \mathbf{h})$$

où

$$R = \exp(\tilde{\mathbf{v}})$$

et

$$\mathbf{b}(\mathbf{v}, \mathbf{h}) = \frac{\|\mathbf{v}\| - \sin \|\mathbf{v}\|}{\|\mathbf{v}\|^3} \langle \mathbf{v}, \mathbf{h} \rangle \mathbf{v} + \frac{\sin \|\mathbf{v}\|}{\|\mathbf{v}\|} \mathbf{h} + \frac{1 - \cos \|\mathbf{v}\|}{\|\mathbf{v}\|^2} \mathbf{h} \wedge \mathbf{v}$$

2. $\exp(\tilde{\mathbf{v}})$ est alors la valeur $\gamma(1)$ de la courbe intégrale γ du champ invariant $R\tilde{\mathbf{v}}$ et de condition initiale $\gamma(0) = Id$.

3. $\exp(\tilde{\mathbf{v}})$ est alors la valeur $\gamma(1)$ de la géodésique γ de condition initiale $\gamma(0) = Id$ et $\gamma'(0) = \tilde{\mathbf{v}}$.

En identifiant $R\tilde{\mathbf{b}}(\mathbf{v}, \mathbf{h})$ et $R\tilde{\Omega}$ nous en tirons \mathbf{h}

$$\mathbf{h} = (d\exp(\tilde{\mathbf{v}}))^{-1}.R\tilde{\Omega} = \mathbf{a}(\mathbf{v}, \Omega)$$

où

$$\mathbf{a}(\mathbf{v}, \Omega) = \Omega + \frac{1}{2}\mathbf{v} \wedge \Omega + \left(\frac{1}{\|\mathbf{v}\|^2} - \frac{\sin \|\mathbf{v}\|}{2\|\mathbf{v}\|(1 - \cos \|\mathbf{v}\|)} \right) \mathbf{v} \wedge (\mathbf{v} \wedge \Omega)$$

En reprenant le système C.3 on obtient ainsi une “bonne” méthode d’Euler à l’aide du schéma suivant :

$$\begin{cases} \mathbf{v}_0 & = 0 \\ R_i & = R_0 \exp(\tilde{\mathbf{v}}_i) \\ \mathbf{v}_{i+1} & = \mathbf{v}_i + \frac{1}{n} \mathbf{a}(\mathbf{v}_i, \Omega(\frac{i}{n})) \end{cases}$$

Nous avons translaté la paramétrisation autour de R_0 , mais nous pouvons tout aussi bien la recentrer autour de R_i à chaque étape. Puisque $\mathbf{a}(\mathbf{0}, \Omega) = \Omega$, il apparaît qu’il est **équivalent d’utiliser la méthode d’Euler centrée à chaque pas ou de considérer Ω constant sur chaque intervalle $[\frac{i}{n}, \frac{i+1}{n}]$** . En effet, l’équation $\mathcal{R}' = R\tilde{\Omega}(\frac{i}{n})$ de condition initiale R_i admet pour solution

$$R(s) = R_i \exp(s\tilde{\Omega}(\frac{i}{n}))$$

Nous avons implémenté les deux méthodes d’Euler (centrée ou non) et nous n’avons pas observé de différence sensible⁴. Nous avons conservé la méthode centrée à chaque étape qui requiert moins de calcul. Nous avons ainsi remplacé l’équation de propagation C.2 par :

$$R_{i+1} = R_i \exp(\frac{1}{n} \tilde{\Omega}(\frac{i}{n}))$$

4. D’une manière générale, l’influence de la paramétrisation employée pour la méthode d’Euler n’intervient qu’au deuxième ordre.

Annexe D

Interpolation des repères (seconde version)

Dans ce qui suit le terme “converge” doit être compris dans le sens de la définition 2.3 de la section 2.1.3. C’est une façon condensée de dire que l’erreur $\|y_i^n - y(\frac{i}{n})\|$, entre une suite de séquences $(y_i^n)_{0 \leq i \leq n}$ et une fonction y , converge uniformément vers 0.

Nous devons vérifier que le schéma d’interpolation de $R_{t,i}^n$:

$$\begin{cases} R_{t,0}^n = R_{t,0} \\ R_{t,i+1}^n = R_{t,i}^n r_{t,i}^n \end{cases} \quad (\text{D.1})$$

où $r_{t,i}^n$ est le point de paramètre t sur la géodésique (de $SO(3, \mathbb{R})$) joignant $r_{0,i}^n$ et $r_{1,i}^n$, converge vers la solution du système

$$\begin{cases} R_t(0) = R_{t,0} \\ R_t'(s) = R_t(s) \tilde{\Omega}_t(s) \end{cases} \quad (\text{D.2})$$

où Ω_t est une interpolation linéaire des limites Ω_0 et Ω_1 de $\Omega_{0,i}^n = n(r_{0,i}^n - Id)$ et de $\Omega_{1,i}^n = n(r_{1,i}^n - Id)$.

Pour cela nous montrerons que

1. $\Omega_{0,i}^n$ et $\Omega_{1,i}^n$ converge effectivement vers des VVRI Ω_0 et Ω_1 lorsqu’on utilise les constructions du repère de Frenet et du RTRP de la section 2.2.2,
2. que $\Omega_{t,i}^n = n(r_{t,i}^n - Id)$ converge alors vers $\Omega_t = (1 - t) \Omega_0 + t \Omega_1$,
3. et que, dans ce cas, le schéma D.1 converge bien vers une solution de D.2.

Nous utiliserons le fait que (c.f. par exemple [MT86])

$$\exp(\tilde{\mathbf{v}}) = Id + \tilde{\mathbf{v}} + o(\|\tilde{\mathbf{v}}\|)$$

où \exp est l'exponentielle de matrice, et que

$$\log(Id + r) = r + o(\|r\|)$$

où \log est l'inverse local de \exp .

Démonstration du point 1 : pour le RTRP nous avons par construction :

$$r_i^n = \exp(\alpha_i^n \tilde{\mathbf{b}}_i^n)$$

d'où

$$n(r_i^n - Id) = n\alpha_i^n \tilde{\mathbf{b}}_i^n + o(n\alpha_i^n \tilde{\mathbf{b}}_i^n)$$

Or nous avons vu section 2.2.3 que cette expression converge bien vers le VVRI du RTRP.

Pour le repère de Frenet, écrivons Ω_i^n sous la forme

$$\Omega_i^n = n(r_i^n - Id) = (R_i^n)^{-1} \frac{R_{i+1}^n - R_i^n}{1/n}$$

Comme $(R_i^n)^{-1}$ converge vers R^{-1} , il suffit de montrer que $n(R_{i+1}^n - R_i^n)$ converge vers $R' = R\tilde{\Omega}$. Ou encore, puisque

$$n(R_{i+1}^n - R_i^n) = n(R_{i+1}^n - R(\frac{i+1}{n})) - n(R_i^n - R(\frac{i}{n})) + n(R(\frac{i+1}{n}) - R(\frac{i}{n}))$$

que la convergence R_i^n est en $o(\frac{1}{n})$. Nous laissons le lecteur s'assurer de cette dernière affirmation \square

Démonstration du point 2 : par définition de $r_{t,i}^n$ et des géodésiques de $SO(3, \mathbb{R})$ nous avons :

$$r_{t,i}^n = r_{0,i}^n \exp(t \log((r_{0,i}^n)^{-1} r_{1,i}^n))$$

Or

$$(r_{0,i}^n)^{-1} r_{1,i}^n = Id + (r_{0,i}^n)^{-1} \frac{\Omega_{0,i}^n - \Omega_{1,i}^n}{n}$$

d'où, en utilisant les approximations d'ordre 1 du logarithme et de l'exponentielle :

$$\begin{aligned} \Omega_{t,i}^n &= n(r_{0,i}^n \exp(t \log((r_{0,i}^n)^{-1} r_{1,i}^n)) - Id) \\ &= n(r_{0,i}^n \exp(t (r_{0,i}^n)^{-1} \frac{\Omega_{0,i}^n - \Omega_{1,i}^n}{n} + o(\frac{1}{n}))) - Id \\ &= n(r_{0,i}^n (Id + t (r_{0,i}^n)^{-1} \frac{\Omega_{0,i}^n - \Omega_{1,i}^n}{n} + o(\frac{1}{n}))) - Id \\ &= n(r_{0,i}^n - Id) + t(\Omega_{0,i}^n - \Omega_{1,i}^n) + no(\frac{1}{n}) \\ &= \Omega_{0,i}^n + t(\Omega_{0,i}^n - \Omega_{1,i}^n) + o(1) \end{aligned}$$

\square

Démonstration du point 3 : par définition de $\Omega_{t,i}^n$, l'équation de propagation du système D.1 se met sous la forme

$$R_{t,i+1}^n = R_{t,i}^n + \frac{1}{n} R_{t,i}^n \tilde{\Omega}_{t,i}^n$$

Notons M_i^n l'approximation par la méthode d'Euler du système D.2 :

$$M_{i+1}^n = M_i^n + \frac{1}{n} M_i^n \tilde{\Omega}_t(\frac{i}{n})$$

Soustrayons ces deux équations :

$$R_{t,i+1}^n - M_{i+1}^n = R_{t,i}^n - M_i^n + \frac{1}{n}(R_{t,i}^n - M_i^n) \tilde{\Omega}_{t,i}^n + \frac{1}{n} M_i^n (\tilde{\Omega}_{t,i}^n - \tilde{\Omega}_t(\frac{i}{n}))$$

En posant

$$e_i^n = \|R_{t,i}^n - M_i^n\|$$

et

$$\epsilon^n = \sup_i \|\tilde{\Omega}_{t,i}^n - \tilde{\Omega}_t(\frac{i}{n})\|$$

et

$$K = \sup_{i,n} \|\tilde{\Omega}_{t,i}^n\|$$

nous voyons par l'inégalité triangulaire que

$$e_{i+1}^n \leq e_i^n (1 + \frac{K}{n}) + \frac{\epsilon_n}{n}$$

Un calcul classique en analyse numérique [Bar77] montre que

$$e_i^n \leq e_0^n \exp(K) + \epsilon_n \frac{\exp(K) - 1}{K}$$

Comme e_0^n est nul et que ϵ_n tend vers 0 d'après le point 2, on en déduit le résultat cherché (on utilise le fait que M_i^n converge vers la solution de D.2) \square

Annexe E

La topologie en tout ou rien

Dans [Mat88] on ne s'intéresse en fait qu'à \mathcal{F} (ou \mathcal{G}) et non à l'ensemble des compacts (pour des raisons que j'ignore...). Nous présentons donc ici la topologie sur \mathcal{F} . Cette topologie, appelée topologie en tout ou rien¹ est définie de la manière suivante :

1. on définit des ensembles de fermés, notés $\mathcal{F}_K^{G_1, \dots, G_n}$ à partir d'une famille finie quelconque d'ouverts G_1, \dots, G_n et d'un compact quelconque K : c'est l'ensemble des fermés de \mathcal{F} qui rencontrent chacun des G_i et évitent K (c.f. figure E.1),
2. ces ensembles $\mathcal{F}_K^{G_1, \dots, G_n}$ sont par définition des ouverts de \mathcal{F} et la topologie en tout ou rien est la topologie engendrée par ces ouverts (c.f. [Bre83] p. 33).

Cette topologie est compacte et séparable, une base dénombrable d'ouverts étant fournie par les ensembles $\mathcal{F}_B^{B_1, \dots, B_n}$, où B, B_1, \dots, B_n sont dans une base dénombrable d'ouverts relativement compacts (i.e d'adhérence compacte) de E (qui est par hypothèse localement compact et séparable).

Convergence dans \mathcal{F} Dans \mathcal{F} , que nous supposons désormais muni de sa topologie, la convergence d'une suite (F_i) de fermés vers un fermé F peut se caractériser par l'une des conditions suivantes :

1. tout ouvert rencontrant F rencontre tous les F_i à partir d'un certain rang et tout compact évitant F évite tous les F_i à partir d'un certain rang,
2. l'ensemble des limites et des valeurs d'adhérences des suites $(x_i), x_i \in F_i$, coïncident et s'identifient à F ,
3. si de plus E est un espace métrique, pour tout x de E , $d(x, F_i)$ converge et vaut $d(x, F)$.

1. le terme anglais *hit or miss topology* est plus parlant puisque, intuitivement, deux fermés sont proches s'ils *rencontrent* les mêmes ouverts et *évitent* les mêmes compacts.

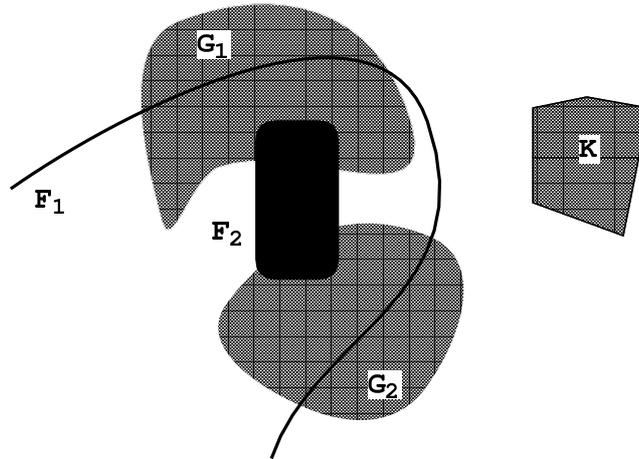


FIG. E.1 – Les fermés F_1 et F_2 sont dans l'ouvert $\mathcal{F}_K^{G_1, G_2}$ défini par les ouverts G_1 et G_2 et le compact K .

Limites supérieures et inférieures On définit alors la limite supérieure $\overline{\text{Lim}} F_i$ d'une suite quelconque (F_i) de fermés par l'une des deux caractérisations équivalentes suivantes :

1. $\overline{\text{Lim}} F_i$ est le plus petit fermé tel que tout compact l'évitant évite tous les F_i à partir d'un certain rang,
2. $\overline{\text{Lim}} F_i$ est l'ensemble des valeurs d'adhérences des suites $(x_i), x_i \in F_i$.

On définit de même la limite inférieure $\underline{\text{Lim}} F_i$ d'une suite quelconque (F_i) de fermés par l'une des deux caractérisations équivalentes suivantes :

1. $\underline{\text{Lim}} F_i$ est le plus grand fermé tel que tout ouvert le rencontrant rencontre tous les F_i à partir d'un certain rang,
2. $\underline{\text{Lim}} F_i$ est l'ensemble des limites des suites convergentes $(x_i), x_i \in F_i$.

On note² alors qu'une suite (F_i) de fermés converge vers un fermé F si et seulement si

$$\underline{\text{Lim}} F_i = \overline{\text{Lim}} F_i = F$$

2. Dans [SMa88] on définit la notion de treillis complet comme un ensemble partiellement ordonné tel que toute sous-famille finie ou non admet un plus petit majorant (le sup) et un plus grand minorant (l'inf) pour la relation d'ordre. On note ensuite que \mathcal{F} forme un treillis complet pour l'inclusion avec

$$\begin{aligned} \sup F_i &= \overline{\bigcup_i F_i}, \\ \inf F_i &= \bigcap_i F_i \end{aligned}$$

Semi-continuité Les limites sup et inf étant définies on dira qu'une fonction f d'un espace topologique séparable E' dans \mathcal{F} est *semi-continue supérieurement* si l'une des propriétés équivalentes suivantes est vérifiée pour toute suite de E' convergente, $x_i \mapsto x$:

1. ou bien $\overline{\text{Lim}} f(x_i) \subset f(x)$,
2. ou bien tout compact évitant $f(x)$ évite tous les $f(x_i)$ à partir d'un certain rang,
3. ou bien toute valeur d'adhérence d'une suite $(y_i), y_i \in f(x_i)$ est dans $f(x)$.

Ce qui équivaut à dire que pour tout compact de E , l'image réciproque des fermés qui l'évitent est un ouvert de E' .

On définit de même la *semi-continue inférieure*. f est semi-continue inférieurement si l'une des propriétés équivalentes suivantes est vérifiée pour toute suite de E' convergente, $x_i \mapsto x$:

1. ou bien $\underline{\text{Lim}} f(x_i) \supset f(x)$,
2. ou bien tout ouvert rencontrant $f(x)$ rencontre tous les $f(x_i)$ à partir d'un certain rang,
3. ou bien tout point de $f(x)$ est limite d'une suite $(y_i), y_i \in f(x_i)$.

Ce qui équivaut à dire que pour tout ouvert de E , l'image réciproque des fermés qui le rencontrent est un ouvert de E' .

On s'attend alors à retrouver les définitions classiques des limite sup et inf d'une suite (x_i) :

$$\overline{\text{lim}} x_i = \inf_i \sup_{j \geq i} x_j,$$

$$\underline{\text{lim}} x_i = \sup_i \inf_{j \geq i} x_j$$

Or seule la première relation est juste dans notre cas (i.e. $\overline{\text{Lim}} F_i = \bigcap_i \overline{\bigcup_{j \geq i} F_j}$). Ces définitions sont donc propres à la morphologie mathématique et l'on doit se garder de toute généralisation...

Annexe F

Construction d'un champ sur un triangle

Nous donnons un exemple de construction d'un champ de vecteurs sur un triangle quelconque, ne s'annulant qu'aux sommets du triangle et tangent aux arêtes orientées (de manière non cyclique) du triangle. La définition de ce champ doit être telle qu'en recollant deux triangles le long d'une arête commune et de même orientation, leurs champs se recollent continûment.

Nous procédons en deux étapes. En premier lieu nous définissons un champ sur un triangle particulier, qui est nul au sommet de ce triangle et tangent à ses arêtes. Puis, dans un second temps, nous "renormalisons" ce champ sur un triangle équilatéral. De cette façon nous pouvons définir un champ convenable sur un triangle quelconque par une simple affinité. Par construction les champs ainsi définis sur des triangles adjacents se recolleront continûment.

F.1 Construction d'un champ particulier

Nous allons construire un champ sur un triangle à partir d'un champ très simple défini sur un secteur D de \mathbb{R}^2 . Pour cela nous allons identifier \mathbb{R}^2 avec le complémentaire de la droite à l'infini dans le plan projectif \mathcal{P}^2 , ce complémentaire pouvant en effet être muni d'une structure d'espace affine.

Puis, par un changement de coordonnées projectives, nous "ramenons" dans le plan affine une portion de la droite à l'infini contenant les directions des droites internes au secteur D , afin de transformer ce secteur (et son champ) en un triangle Δ .

Soit H le champ :

$$H(x, y) = \begin{pmatrix} y \\ x \end{pmatrix}$$

défini sur le secteur D de \mathbb{R}^2 :

$$\begin{cases} x > y \\ x > -y \end{cases}$$

H est un champ tangent aux branches d'hyperboles $x^2 - y^2 = k^2$ qui constituent un feuilletage de D (voir figure F.1). L'immersion canonique $(x, y) \mapsto (x, y, 1)$ de \mathbb{R}^2 dans le plan projectif¹

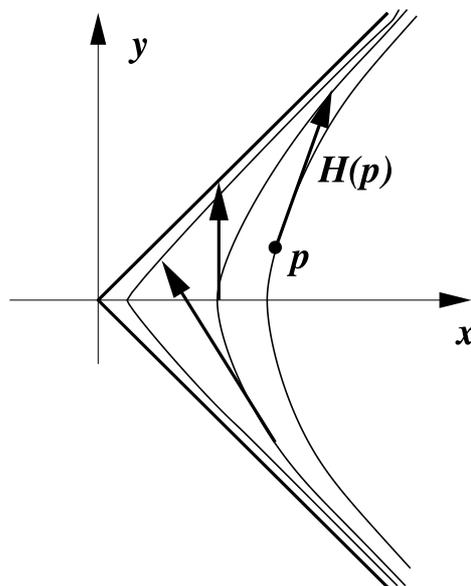


FIG. F.1 – H est un champ tangent aux branches d'hyperboles $x^2 - y^2 = k^2$.

et le changement des coordonnées projectives :

$$(x, y, 1) \mapsto (x, y, x)$$

permettent par reprojection sur le plan $\Pi(x + z = 1)$ de définir une application de D dans Π (voir figure F.2) :

$$(x, y) \mapsto \left(\frac{x}{x+1}, \frac{y}{x+1}, \frac{x}{x+1} \right)$$

En munissant Π de la base $\left(\begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \right)$ on obtient un difféomorphisme ϕ de D dans

1. Pour nous, le plan projectif est donc le quotient de \mathbb{R}^3 privé de 0 par la relation de proportionnalité.

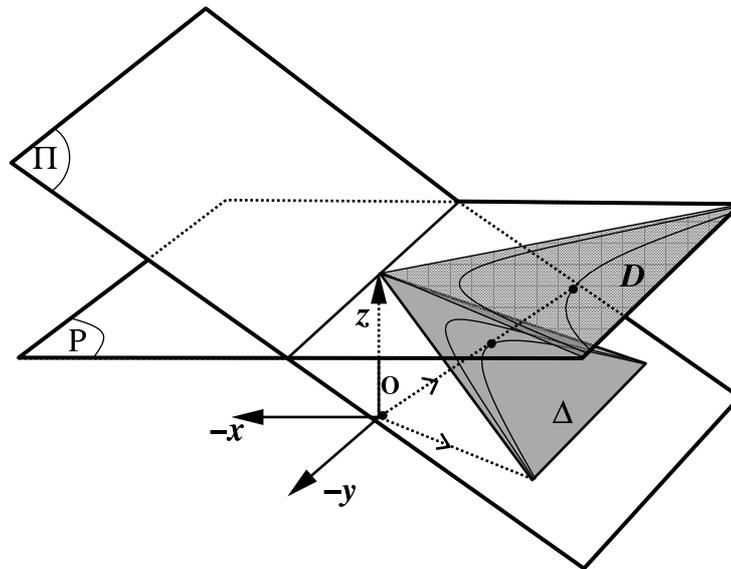


FIG. F.2 – Changement de coordonnées projectives.

l'intérieur du triangle $\Delta = ((1, -1), (1, 1), (0, 0))$:

$$\begin{aligned} \phi : D &\longrightarrow \Delta \\ (x, y) &\longmapsto \left(\frac{x}{x+1}, \frac{y}{x+1} \right) \end{aligned}$$

ϕ transforme le champ H en un champ image ϕ^*H avec :

$$\phi^*H \circ \phi(x, y) = d\phi.H(x, y) = \frac{1}{(1+x)^2} \begin{pmatrix} y \\ x^2 - y^2 + x \end{pmatrix}$$

ou encore, puisque $\phi^{-1}(u, v) = \left(\frac{u}{1-u}, \frac{v}{1-u} \right)$:

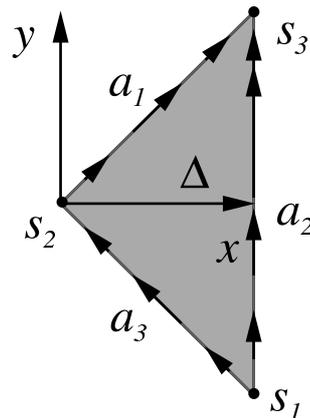
$$\phi^*H(u, v) = \begin{pmatrix} v(1-u) \\ u - v^2 \end{pmatrix}$$

Ce champ se prolonge continûment sur les bords de Δ (voir figure F.3)

$$\left\{ \begin{array}{l} \phi(u, -u) = u(1-u) \begin{pmatrix} -1 \\ 1 \end{pmatrix} \text{ sur } a_1 \\ \phi(1, v) = (1-v^2) \begin{pmatrix} 0 \\ 1 \end{pmatrix} \text{ sur } a_2 \\ \phi(u, u) = u(1-u) \begin{pmatrix} 1 \\ 1 \end{pmatrix} \text{ sur } a_3 \end{array} \right.$$

et ne s'annule qu'aux sommets de Δ .

La droite à l'infini est constituée, par définition, des classes de points dont la troisième coordonnée est nulle.

FIG. F.3 – Le champ se prolonge continûment sur les bords de Δ ...

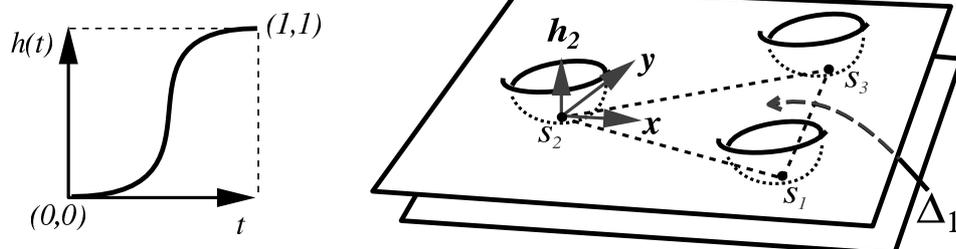
F.2 Renormalisation

Par une affinité A on envoie Δ sur un triangle équilatéral Δ_1 et le champ ϕ^*H sur le champ $A.\phi^*H$. On renormalise alors ce champ sur Δ_1 privé de ses sommets, en le divisant par sa norme. On note H_1 le champ renormalisé.

Soit alors h une fonction C^∞ de $[0,1]$ dans $[0,1]$ telle que $f(0) = 0, f(1) = 1$ et dont les dérivées sont nulles à tout ordre en 0 et 1. Sur le disque centré en $(0,0)$ de rayon $1/3$ on définit h_1 par :

$$h_1(p) = h(\|3p\|)$$

Puis on définit h_2 sur \mathbb{R}^2 par translation de h_1 aux trois sommets de Δ_1 en la prolongeant par 1 (c.f. figure F.4). On pose enfin,

FIG. F.4 – À gauche : le profil de h . À droite : le graphe de h_2 .

$$H_2 = h_{2|\Delta_1} \cdot H_1$$

H_2 (voir figure F.5) est C^∞ sur Δ_1 , en particulier au niveau des trois sommets. On obtient

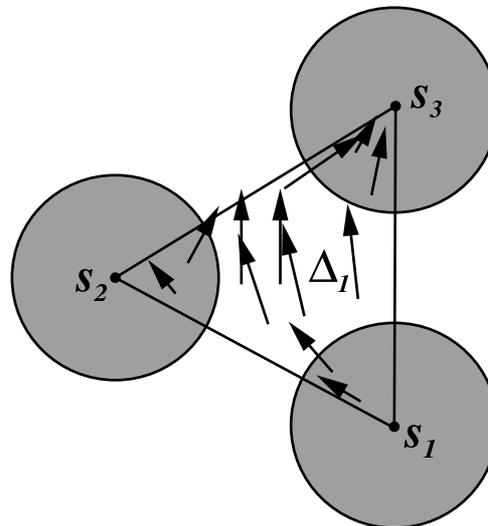


FIG. F.5 – Le champ H_2 .

le champ cherché sur un triangle quelconque par affinité du triangle Δ_1 et de son champ H_2 . Il est clair que l'on peut raccorder continûment les champs ainsi définis sur des triangles adjacents.

Lexique

Ce lexique comprend la plupart des termes mathématiques employés dans ce mémoire. Nous nous sommes contentés de recopier des définitions extraites d'ouvrages classiques en indiquant chaque fois une référence précise. Les mots en *italique* sont eux-mêmes définis dans ce lexique.

Application tangente [BG87], §2.5.13. Soient X et Y deux *variétés* de dimension d et e , f un C^p -morphisme de X dans Y , x un point de X . On appelle application tangente à f en x , l'application linéaire $T_x f$ de l'espace tangent $T_x X$ dans $T_{f(x)} Y$ définie par :

$$T_x f = \eta_{f(x)}^{-1} \circ (\psi \circ f \circ \varphi^{-1})'(\varphi(x)) \circ \theta_x$$

où θ_x et $\eta_{f(x)}$ sont les isomorphismes entre $T_x X$ et \mathbb{R}^d et $T_{f(x)} Y$ et \mathbb{R}^e associés à des *cartes* (U, φ) et (V, ψ) .

Atlas [BG87], §2.2.1. Soit X un ensemble et p un entier strictement positif. On appelle atlas d -dimensionnel de classe C^p sur X , un ensemble de couples $(U_i, \varphi_i)_{i \in I}$, appelés *cartes*, tels que les 3 axiomes suivants soient satisfaits :

1. Les U_i sont des sous ensembles de X et $\bigcup_{i \in I} U_i = X$.
2. Chaque φ_i est une bijection de U_i sur un ouvert de \mathbb{R}^d .
3. Pour tout couple (i, j) l'application $\varphi_j \circ \varphi_i^{-1}$ est un isomorphisme de classe C^p de $\varphi_i(U_i \cap U_j)$ dans $\varphi_j(U_i \cap U_j)$.

Carte Voir *Atlas*.

Champ de vecteurs On appelle champ de vecteurs sur une *variété* X une *section* de son *fibré tangent* TX .

Connexion linéaire [Die71a], §17.18.1. On appelle connexion (linéaire) sur une variété X un X -morphisme (c.f. *espace fibré*) de $TX \oplus TX$ dans TTX , le *fibré tangent* du fibré tangent, vérifiant des conditions de linéarité et de projections évidentes relativement aux diverses fibrations mises en jeu (c.f. [Die71a], §17.16.3. pour plus de détails). À toute connexion on peut associer une unique *dérivation covariante* vérifiant les conditions 1,2 et 3 de la définition de *dérivation covariante* d'une variété riemannienne. On dit que cette dérivation est sans torsion si elle vérifie de plus la condition 4, et qu'elle est compatible avec une métrique sur X si elle vérifie de plus la condition 5.

Crochet de Lie Voir *dérivation suivant un champ de vecteurs*.

Dérivation covariante [Mal72], §III.8.4. Soit X une variété riemannienne de classe C^2 , alors il existe un unique opérateur ∇ , appelé dérivation covariante sur X , qui associe à tout couple A, B de champs de vecteurs de classe C^1 un champ de vecteurs $\nabla_A B$ de classe C^0 et qui vérifie les propriétés suivantes :

1. $\nabla_A B$ est linéaire en A et linéaire en B
2. pour toute fonction réelle sur X , $\nabla_{fA} B = f \nabla_A B$
3. et $\nabla_A f B = f \nabla_A B + (Af)B$ où Af est la *dérivation* de f suivant A
4. $\nabla_A B - \nabla_B A = [A, B]$ où $[\cdot, \cdot]$ est le *crochet de Lie* de A et B .
5. $A\langle B, C \rangle = \langle \nabla_A B, C \rangle + \langle B, \nabla_A C \rangle$ où $\langle \cdot, \cdot \rangle$ désigne le tenseur métrique sur X .

Dérivation suivant un champ de vecteurs [Die71a], §17.14.1. Soit X une variété, A un champ de vecteurs sur X et $f : X \rightarrow \mathbb{R}$ une application différentiable, on appelle dérivation de f suivant le champ A , l'application $Af : X \rightarrow \mathbb{R}$:

$$x \mapsto Af(x) = df(x).A$$

où $df(x)$ est la *différentielle* de f au point x .

L'opérateur $f \mapsto Af$ s'annule si f est constante et vérifie la relation classique de dérivation d'un produit :

$$A(fg) = (Af)g + f(Ag)$$

Inversement on montre qu'on peut associer à tout opérateur vérifiant ces deux propriétés un unique champ de vecteurs dont il est la dérivation.

On définit alors le crochet de Lie de deux champs de vecteurs A et B comme le champ de vecteurs $[A, B]$ associé à la dérivation $f \mapsto A(Bf) - B(Af)$.

Difféomorphisme Un C^p -difféomorphisme est un C^p -morphisme inversible dont l'inverse est également un C^p -morphisme.

Différentielle [BG87], §2.5.23. Soit X une variété, E un espace vectoriel de dimension finie et f un C^p -morphisme de X dans E . On appelle différentielle de f en x , et on note $df(x)$, l'application $\theta_{f(x)} \circ T_x f$ de l'espace tangent $T_x X$ dans E , où $T_x f$ est l'application tangente de f en x et $\theta_{f(x)}$ est l'isomorphisme canonique $T_{f(x)} E \rightarrow E$.

Espace fibré [Die71a] §16.12.1. On appelle espace fibré ou fibration un triplet (X, B, π) , où X et B sont des variétés, π est une *submersion* surjective et vérifie la condition suivante dite de trivialité locale :

- Pour tout $b \in B$, il existe un voisinage ouvert U de b dans B , une variété F et un *difféomorphisme*

$$\varphi : U \times F \rightarrow \pi^{-1}(U)$$

tel que $\pi(\varphi(y, t)) = y$ quels que soient $y \in U$ et $t \in F$.

On dit que X est l'espace de fibration, B sa base et π sa projection ; pour tout $b \in B$, $\pi^{-1}(b)$ est une sous-variété fermée appelée fibre (au dessus) de b .

Lorsque les fibres sont des espaces vectoriels de dimension finie et que φ est linéaire en restriction à chaque fibre, on dit que le triplet (X, B, π) est un (espace) fibré vectoriel. Si E et F sont deux fibrés vectoriels de même base B , on définit la somme directe

$$E \oplus F$$

comme le fibré vectoriel de base B dont la fibre au dessus de $b \in B$ est la somme directe des fibres de E et F au dessus de b .

De plus, on appelle section d'une fibration (X, B, π) , une application s de la base B dans X telle que :

$$\pi \circ s = Id_B.$$

Si (X, B, π) et (Y, B, p) sont deux fibrations de même base B , on appelle B-morphisme un *morphisme* de X dans Y vérifiant

$$\pi = p \circ f$$

Pour une bonne introduction aux espaces projetés et aux espaces fibrés, voir [Pha92], chapitre VI.

Espace tangent [BG87], §2.5.9. Soient X une *variété*, de dimension d , de classe C^p , x un point de X , (U, φ) et (V, ψ) deux *cartes* de X en x . Soient u et v deux éléments de \mathbb{R}^d . La relation \mathcal{R} , définie entre les triplets (U, φ, u) et (V, ψ, v) par :

$$(U, \varphi, u) \mathcal{R} (V, \psi, v) \text{ si et seulement si } (\varphi \circ \psi^{-1})'(\psi(x))(u) = v$$

est une relation d'équivalence.

On appelle vecteur tangent à X en x une classe d'équivalence de \mathcal{R} . On note $T_x X$, et on appelle espace tangent à X en x , l'ensemble des vecteurs tangents à X en x .

Fibré tangent [BG87], §2.5.25. Soit X une *variété* de dimension d et de classe C^p . La réunion des *espaces tangents* $T_x X$ lorsque x parcourt X peut être canoniquement muni d'une structure de variété de dimension $2d$ et de classe C^{p-1} . Cette variété est appelé le fibré tangent de X .

TX est un *espace fibré* vectoriel de base X et de fibres $T_x X$.

Groupe de Lie [Mal72], §III.1.1. On appelle structure de groupe de Lie sur G , la donnée :

1. d'une structure de *variété* de classe C^3 sur G ,
2. et d'une structure de groupe sur G

reliées par la condition suivante de compatibilité : le produit

$$\begin{aligned} G \times G &\rightarrow G \\ (g, g') &\mapsto gg' \end{aligned}$$

et l'inverse

$$\begin{aligned} G &\rightarrow G \\ g &\mapsto g^{-1} \end{aligned}$$

sont des C^3 -*morphismes*.

Notons que l'on remplace souvent la condition C^3 par une condition C^∞ ([Die71a]).

Homéomorphisme *Morphisme* continu et inversible dont l'inverse est également continu.

Homotopie [Die71a], §16.26.1. Soient f et g deux applications continues d'un espace topologique X dans un espace topologique Y . Une homotopie de f dans g est une application continue φ de $X \times [0, 1]$ dans Y telle que :

$$\varphi(x, 0) = f(x) \text{ et } \varphi(x, 1) = g(x).$$

Lorsqu'une telle homotopie existe, on dit que f est homotope à g .

Image d'un champ de vecteur [Pha92], §III.2. Soient v un *champ de vecteurs* sur une *variété* X et ϕ un *difféomorphisme* de X sur une *variété* Y . On appelle image du champ de vecteurs v par le difféomorphisme ϕ le champ de vecteurs ϕ^*v défini sur Y par :

$$\phi^*v(\tilde{x}) = T_x\phi.v(x), \text{ où } x = \phi^{-1}(\tilde{x}).$$

Immersion [BG87], §2.6.9. Soient X et Y deux *variétés* et $f : X \rightarrow Y$ un *morphisme*. On dit que f est une immersion en x si l'*application tangente* T_xf est injective. On dit que f est une immersion si la propriété est vraie pour tout x de X .

Morphisme, Morphisme de groupe de Lie [BG87], §2.3.1. Soient X et Y deux *variétés* de dimension d et e et de classe C^q et C^r . Soit $p \leq \inf(q, r)$. On dit qu'une application continue f de X dans Y est un C^p -morphisme, si pour toute *carte* (U, φ) en x , et pour toute *carte* (V, ψ) de Y en $f(x)$, l'application $\psi \circ f \circ \varphi^{-1}$ est de classe C^p de $\phi(U \cap f^{-1}(V))$ dans \mathbb{R}^e .

Si X et Y sont des *groupes de Lie*, on dit qu'une application f de X dans Y est un C^p -morphisme de groupe de Lie, si f est un C^p -morphisme pour les variétés sous-jacentes à X et Y et si f est un morphisme de groupe pour les groupes sous-jacents à X et Y .

Opération (ou action) d'un groupe [Fer88], §I.1. Soit G un groupe de loi \cdot et d'élément neutre e . On appelle opération à gauche de G sur un ensemble X une application $F : G \times X \rightarrow X$ vérifiant les conditions suivantes :

1. $\forall x \in X, F(e, x) = x$.
2. $\forall h, g \in G, \forall x \in X, F(g, F(h, x)) = F(g.h, x)$

On dit que cette opération est transitive si

$$\forall x \in X, F(G, x) = E$$

On dit qu'elle est simplement transitive si

$$\forall x, y \in X, \exists !g \in G, F(g, x) = y$$

[Die71a], §16.10.1. Si G est un *groupe de Lie* et X une *variété*, on dit que G opère différemmentiellement sur X si l'on s'est donné une opération à gauche de G sur X qui est une application C^∞ de $G \times X$ dans X .

Plongement [BG87], §2.6.9. Soient X et Y deux *variétés* et $f : X \rightarrow Y$ un *morphisme*. On dit que f est un plongement si elle est : injective, une *immersion* et un *homéomorphisme* de X sur son image $f(X)$.

Revêtement [BG87], §2.4.1. Soient X et Y deux *variétés* de classe C^s . Une application p de X dans Y est dite un revêtement si les propriétés suivantes sont vérifiées :

1. p est surjective et est un *morphisme* de classe C^s de X dans Y ,
2. pour tout y de Y , il existe un ouvert V de Y contenant y , tel que $p^{-1}(V)$ admette une partition de la forme $p^{-1}(V) = \cup_{i \in I} U_i$ où les U_i sont des ouverts de X , tels que pour tout i la restriction de p à U_i soit un *difféomorphisme* de U_i sur V .

On peut également voir un revêtement comme un *espace fibré* (Y, X, p) dont les fibres sont discrètes [Pha92], §VII.2.

Section d'un espace fibré voir *espace fibré*.

Sous-variété de \mathbb{R}^n [BG87], §2.2.1. Soit V une partie de \mathbb{R}^n . On dit que V est une sous-variété de \mathbb{R}^n , de dimension d , de classe C^p , si pour tout x de V , il existe un ouvert U de \mathbb{R}^n , contenant x , un difféomorphisme f de classe C^p de U sur son image $f(U)$ ouvert de \mathbb{R}^n , tel que $f(U \cap V) = f(U) \cap \mathbb{R}^d$. L'entier $n - d$ est alors appelé la **codimension** de V .

Submersion [BG87], §2.6.9. Soient X et Y deux *variétés* et $f : X \rightarrow Y$ un *morphisme*. On dit que f est une submersion en x si l'*application tangente* $T_x f$ est surjective. On dit que f est une submersion si la propriété est vraie pour tout x de X .

Variété [BG87], §2.2.5. Une variété (ou variété différentielle) de dimension d de classe C^p ($p \geq 1$) est la donnée d'un ensemble X et d'une classe d'équivalence d'*atlas* d -dimensionnels de classe C^p sur X .

Variété riemannienne [Mal72] §III.8.1.2. On appelle champ de tenseurs covariants de degré 2, symétrique défini positif sur une *variété* X , la donnée, pour tout $x \in X$, d'une forme bilinéaire symétrique définie positive sur l'*espace tangent* $T_x X$.

On appelle variété riemannienne une variété munie d'un tel champ de tenseurs.

Bibliographie

- [AH80] Appel (K.) et Haken (W.). – La solution du problème des quatre couleurs. *In: Les progrès des mathématiques.* – Belin, Bibliothèque Pour la Science, 1980.
- [Aur90] Aurenhammer (F.). – *Voronoi diagrams: a survey of a fundamental geometric data structure.* – Rapport technique n° B-90-09, Berlin, West Germany, Fachber. Math., Free Univ. Berlin, 1990.
- [Bar77] Baranger (J.). – *Introduction à l'analyse numérique*, chap. 7: Problème de Cauchy pour les équations différentielles ordinaires. – Paris, Hermann, 1977, *Collection méthodes.*
- [Bau75] Baumgart (B.G.). – A polyhedron representation for computer vision. *In: AFIPS Conference*, pp. 589–596.
- [BCGH92] Barr (A.H.), Currin (B.), Gabriel (S.) et Hughes (J.F.). – Smooth interpolation of orientations with angular velocity constraints using quaternions. *Computer Graphics (SIGGRAPH'92)*, vol. 26, n° 2, juillet 1992, pp. 313–320.
- [Ber70] Berge (C.). – *Graphes et hypergraphes.* – Paris, Dunod, 1970.
- [BG87] Berger (M.) et Gostiaux (B.). – *Géométrie différentielle, variétés, courbes et surfaces.* – PUF, 1987.
- [Bis75] Bishop (R.L.). – There is more than one way to frame a curve. *American Mathematical Monthly*, no82, 1975, pp. 246–251.
- [BN92] Beier (T.) et Neely (S.). – Feature-based image metamorphosis. *Computer Graphics (SIGGRAPH'92)*, vol. 26, n° 2, juillet 1992, pp. 35–42.
- [Boi88] Boissonnat (J.-D.). – Shape reconstruction from planar cross sections. *Computer Vision, Graphics and Image Processing*, vol. 44, 1988.
- [Bre83] Brezis (H.). – *Analyse fonctionnelle. Théorie et applications.* – Masson, 1983, collection mathématiques appliquées pour la maîtrise édition.

- [BU89] Bethel (E.W.) et Uselton (S.P.). – Shape distortion in computer-assisted key-frame animation. *In: Computer Animation'89, State-of-the-art in Computer Animation*. Springer-Verlag, pp. 215–224. – Computer Graphics International, 1989.
- [Car93] Carmo (M.P. Do). – *Riemannian geometry*. – Birkhauser, 1993.
- [Cho95] Chou (J.J.). – Voronoi diagrams for planar shapes. *IEEE Computer Graphics and Applications*, vol. 15, n° 2, mars 1995, pp. 52–59.
- [CL92] Chen (X.) et Lienhardt (P.). – Modelling and programming evolutions of surfaces. *Computer graphics forum*, vol. 11, n° 5, 1992, pp. 323–341.
- [CLR94] Cormen (T.), Leiserson (C.) et Rivest (R.). – *Introduction à l'algorithmique*. – Dunod, 1994.
- [CNR92] CNRS (Formation permanente). – *Les Méthodes Non Standard: de la mathématique à la physique*. – Gif sur Yvette, avril 1992.
- [CP89] Chen (S.E.) et Parent (R.E.). – Shape averaging and its applications to industrial design. *IEEE Computer Graphics and Applications*, vol. 9, n° 11, janvier 1989, pp. 47–54.
- [CS78] Christiansen (H.N.) et Sederberg (T.W.). – Conversion of complex contour line definitions into polygonal element mosaics. *Computer Graphics (SIGGRAPH '78)*, vol. 12, n° 3, août 1978, pp. 187–192.
- [CS93] Chopp (D.L.) et Sethian (J.A.). – Flow under curvature: singularity formation, minimal. *Experimental mathematics*, vol. 2, n° 4, 1993, pp. 235–255.
- [Die71a] Dieudonné (J.). – *ELEMENTS D'ANALYSE : 3 : chap. 16 et 17*. – Cahiers scientifiques, 1971.
- [Die71b] Dieudonné (J.). – *ELEMENTS D'ANALYSE : 4 : chap. 18 à 20*. – Cahiers scientifiques, 1971.
- [Die75] Dieudonné (J.). – *ELEMENTS D'ANALYSE : 5 : chap. 21*. – Cahiers scientifiques, 1975.
- [DTG95] Desbrun (M.), Tsingos (N.) et Gascuel (M-P.). – Adaptive sampling of implicit surfaces for interactive modeling and animation. *In: Implicit Surfaces'95*, éd. par Wyvill (B.) et Gascuel (M-P.). Eurographics, pp. 171–185. – Grenoble, avril 1995.

- [Ebe92] Ebel (R.). – Reconstruction interactive d'éléments anatomiques à l'aide de surfaces de forme libre. – Thèse de doctorat en sciences, Ecole Nationale Supérieure des Télécommunications, Paris, décembre 1992.
- [EPO91] Ekoule (A.), Peyrin (F.) et Odet (C.). – A triangulation algorithm for arbitrary shaped multiple planar contours. *ACM Transactions on Graphics*, vol. 10, n° 2, avril 1991, pp. 182–199.
- [Fau89] Faugeras (O.). – A few steps toward artificial 3D vision. *In: Robotics science*, éd. par Brady (Michael), pp. 39–137. – MIT press, 1989.
- [Fau93] Faugeras (O.). – Cartan's moving frame method and its application to the geometry and evolution of curves in the euclidean affine and projective planes. *In: Applications of invariance in computer vision*, éd. par Mundy (J.L.), Zisserman (A.) et Forsyth (D.), pp. 11–46. – Springer-Verlag, 1993.
- [FDFH90] Foley (J.D.), Dam (A. Van), Feiner (S.K.) et Hughes (J.). – *Computer graphics : principles and practice*. – Addison-Wesley, 1990, second edition édition.
- [Fer88] Ferrif (B.A.). – *Méthodes mathématiques. Géométrie*. – Paris, Institut Géographique National, 1988.
- [FKU77] Fusch (H.), Kedem (Z.M.) et Uselton (S.P.). – Optimal surface reconstruction from planar contours. *Communications of the ACM*, vol. 20, n° 10, octobre 1977, pp. 693–702.
- [FN94] Foley (T.A.) et Nielson (G.M.). – Modeling of scattered multivariate data. *In: State of the art reports*. Eurographics'94, pp. 38–59. – Oslo, Norway, septembre 1994.
- [Gas90] Gascuel (M.-P.). – Déformations de surfaces complexes: techniques de haut niveau pour la modélisation et l'animation. – Thèse de doctorat en sciences, Université de Paris-Sud, octobre 1990.
- [GD95] Gelston (S.M.) et Dutta (D.). – Boundary surface recovery from skeleton curves and surfaces. *Computer Aided Geometric Design*, vol. 12, n° 1, 1995, pp. 27–51.
- [Gei93] Geiger (B.). – *Three-dimensional modeling of human organs and its application to diagnosis and surgical planning*. – Rapport technique n° 2105, INRIA, novembre 1993.

- [GG95] Goldstein (E.) et Gotsman (C.). – Polygon morphing using a multiresolution representation. *In: Graphic Interface'95*. – Quebec, mai 1995.
- [GH94] Guibas (L.) et Hershberger (J.). – Morphing simple polygons. *In: 10th Computational geometry*. – Stony brook, NY, USA, juin 1994.
- [Gra71] Gramain (A.). – *Topologie Des Surfaces*. – Collection SUP chez P.U.F, 1971.
- [GS94] Grünbaum (B.) et Shepard (G.C.). – A new look at Euler's theorem for polyhedra. *American Mathematical Monthly*, février 1994, pp. 109–128.
- [GV95] Gomes (J.) et Velho (L.). – Abstraction paradigms for computer graphics. *The Visual Computer*, vol. 11, n° 5, mai 1995, pp. 227–239.
- [GYKD91] Goldak (J.A.), Yu (X.), Knight (A.) et Dong (L.). – Constructing discrete medial axis of 3D objects. *International Journal of Computational Geometry and Applications*, vol. 1, n° 3, 1991, pp. 327–339.
- [Hof89] Hoffmann (C. M.). – *Geometric and Solid Modeling: An Introduction*. – Morgan Kaufmann, 1989.
- [HP84] Hafford (K.J.) et Preston (K.). – Three-dimensional skeletization of elongated solids. *Computer Vision, Graphics, and Image Processing*, vol. 27, 1984, pp. 78–91.
- [HP93] Hanotaux (G.) et Peroche (B.). – Interactive control of interpolations for animation and modeling. *In: Graphic Interface'93*. – Toronto, mai 1993.
- [HP94] Hilton (P.) et Pedersen (J.). – Euler's theorem for polyhedra: a topologist and geometer respond. *American Mathematical Monthly*, décembre 1994, pp. 959–962.
- [Hug92] Hughes (J.F.). – Scheduled Fourier volume morphing. *Computer Graphics (SIGGRAPH'92)*, vol. 26, n° 2, juillet 1992, pp. 43–46.
- [Hus94] Husek (M.). – Introduction to categorical shape theory, with applications in mathematical morphology. *In: Shape in picture*. NATO advanced research workshop, pp. 92–110. – Springer.
- [IKSU92] Ikeda (T.), Kunii (T.L.), Shinagawa (Y.) et Ueda (M.). – A geographical database based on the homotopy model. *In: Modern geometric computing for visualization*, éd. par Kunii (T.L.) et Shinagawa (Y.). pp. 193–206. – Tokyo, Japon, 1992.

- [KCM94] Karron (D.B.), Cox (J.) et Mishra (B.). – New findings from the SpiderWeb algorithm: Towards a digital Morse theory. *In: Visualization in Biomedical Computing '94*, éd. par Robb (R.A.). pp. 643–657. – Rochester, octobre 1994.
- [KCP92] Kent (J.R.), Carlson (W.E.) et Parent (R.E.). – Shape transformation for polyhedral objects. *Computer Graphics (SIGGRAPH'92)*, vol. 26, n° 2, juillet 1992, pp. 47–54.
- [Ker81] Kergosien (Y.L.). – La famille des projections orthogonales d'une surface et ses singularités. *C.R.A.S, t.292, Série I*, juin 1981, pp. 929–932.
- [Ker92] Kergosien (Y.L.). – Topology and visualization: from generic singularities to combinatorial shape modelling. *In: International workshop on modern geometric computing for visualization*, éd. par T.L. Kunii (Y. Shinagawa). – Tokyo, Japan, 1992.
- [KKM91] Kong (Y.), Kopperman (R.) et Meyer (P.). – A topological approach to digital topology. *Amer. Math. Monthly*, vol. 98, n° 10, 1991, pp. 901–917.
- [Klo86] Klok (F.). – Two moving coordinate frames for sweeping along a 3D trajectory. *Computer Aided Geometric Design*, no3, 1986, pp. 217–229.
- [KPC91] Kent (J.R.), Parent (R.E.) et Carlson (W.E.). – Establishing correspondences by topological merging: a new approach to 3-D shape transformation. *In: Graphic Interface'91*, pp. 271–278. – Calgary, juin 1991.
- [KR91] Kaul (A.) et Rossignac (J.). – Solid-interpolating deformations: construction and animation of PIPS. *In: Eurographics'91*, pp. 493–505. – Vienna, septembre 1991.
- [KST92] Kunii (T.L.), Shinagawa (Y.) et Takahashi (S.). – Algorithmic animation of constructing surfaces from cells. *In: Creating and Animating The Virtual World*, éd. par Thalmann (D.). pp. 191–197. – Springer.
- [KTZ92] Kimia (B.B.), Tannenbaum (A.) et Zucker (S.W.). – On the evolution of curves via a function of curvature. I. The classical case. *Journal of Mathematical Analysis and Applications*, vol. 163, 1992, pp. 438–458.
- [KWT87] Kass (M.), Witkin (A.) et Terzopoulos (D.). – Snakes: Active contour models. *In: ICCV*, pp. 259–268.

- [Laz95] Lazarus (F.). – *A Geometrically based approach to 3D skeleton curve blending*. – Rapport technique n° 2458, Rocquencourt, France, INRIA, janvier 1995.
- [LCHS94] Lee (S.Y.), Chwa (K.Y.), Hahn (J.) et Shin (S.Y.). – Image morphing using deformable surfaces. *In: Computer Animation'94*, pp. 31–39.
- [LCJ93] Lazarus (F.), Coquillart (S.) et Jancène (P.). – Interactive axial deformations. *In: Modeling in Computer Graphics*. Springer-Verlag, pp. 241–254. – Genova, juin 1993.
- [LCJ94] Lazarus (F.), Coquillart (S.) et Jancène (P.). – Axial deformations: an intuitive deformation technique. *Computer-Aided Design*, vol. 26, n° 8, août 1994, pp. 607–613.
- [LCSW95] Lee (S.Y.), Chwa (K.Y.), Shin (S.Y.) et Wolberg (G.). – Image metamorphosis using snakes and free-form deformations. *In: Proceedings of SIGGRAPH '95 (Los Angeles, CA)*. ACM SIGGRAPH, pp. 439–448. – ACM Press.
- [LFA77] Lelong-Ferrand (J.) et Arnaudies (J.M.). – *Géométrie*. – T 3, Dunod Université, 1977.
- [LGL95] Leros (A.), Garfinkle (C.) et Levoy (M.). – Feature based volume metamorphosis. *In: Proceedings of SIGGRAPH '95 (Los Angeles, CA)*. ACM SIGGRAPH, pp. 449–456. – ACM Press.
- [Lie93] Lienhardt (P.). – *N-dimensional generalized combinatorial maps and cellular quasi-manifolds*. – Rapport technique n° 93/04, Strasbourg, Université Louis Pasteur, 1993.
- [LL89] Leymarie (F.) et Levine (M.D.). – *Snakes and Skeletons*. – Rapport technique n° TR-CIM-89-3, McGill Research Centre for Intelligent Machines, janvier 1989.
- [LV94] Lazarus (F.) et Verroust (A.). – Feature-based shape transformation for polyhedral objects. *In: Fifth Eurographics Workshop on Animation and Simulation*. – Oslo, Norway, septembre 1994.
- [LV95] Lazarus (F.) et Verroust (A.). – Décomposition cylindrique de polyèdre et courbe squelette. *In: troisième journées de l'AFIG*. – Marseille, novembre 1995.
- [Mal72] Malliavin (P.). – *Géométrie Différentielle intrinsèque*. – Paris, Hermann, 1972, *Enseignement des sciences*.

- [Mat88] Matheron (G.). – Exemples of topological properties of skeletons. *In: Image analysis and mathematical morphology*, éd. par Serra (J.), chap. 11, pp. 217–238. – Academic Press, 1988.
- [MMP87] Mitchell (J.S.B.), Mount (D.M.) et Papadimitriou (C.H.). – The discrete geodesic problem. *SIAM J. Comput.*, vol. 16, 1987, pp. 647–668.
- [MR95] Mayya (N.) et Rajan (V.T.). – An efficient shape representation scheme using voronoi skeletons. *Pattern Recognition Letters*, vol. 16, n° 2, février 1995, pp. 147–160.
- [MSS92] Meyers (D.), Skinner (S.) et Sloan (K.). – Surfaces from contours. *ACM Transactions on Graphics*, vol. 11, n° 3, juillet 1992, pp. 228–258.
- [MT86] Mneimné (R.) et Testard (F.). – *Introduction à la théorie des groupes de Lie classiques*, chap. 3: La fonction exponentielle. Applications. – Paris, Hermann, 1986, *Collection Méthodes*.
- [OP94] Opitz (K.) et Pottmann (H.). – Computing shortest paths on polyhedra: applications in geometric modeling and scientific visualization. *International Journal of Computational Geometry and Applications*, vol. 4, n° 2, 1994, pp. 165–178.
- [Paj92] Pajon (J.L.). – Outils de visualisation interactifs pour l'analyse de données scalaires structurées : applications aux sciences de la terre. – Thèse de doctorat de l'université Paris-Sud, 1992.
- [Par92] Parent (R.E.). – Shape transformation by boundary representation interpolation: a recursive approach to establishing face correspondences. *The Journal of Visualization and Computer Animation*, vol. 3, 1992, pp. 219–239.
- [Ped95] Pedersen (H.K.). – Decorating implicit surfaces. *In: Proceedings of SIGGRAPH '95 (LA, Californie)*, éd. par Cook (R.), pp. 291–300. – ACM Press.
- [Pha92] Pham (F.). – *Géométrie et calcul différentiel sur les variétés*. – Paris, InterEditions, 1992.
- [PLVB94] Paouri (A.), Lazarus (F.), Verroust (A.) et Blonz (C.). – Metadata. – videotape, INRIA UCIS Audiovisuel, France, 1994.
- [Pon88] Ponce (J.). – Représentation des objets tridimensionnels. – Thèse d'Etat de l'université Paris-Sud, 1988.

- [Pre] Preteux (F.). – La morphologie mathématique. ses fondements: ensembliste, topologique, probabiliste. – Cours de TELECOM Paris.
- [Req80] Requicha (A.A.G.). – Representations for rigid solids: Theory methods, and systems. *ACM Computing Surveys*, vol. 12, décembre 1980, pp. 437–464.
- [Rév93] Réveillès (J.P.). – Ideal discrete geometry. *In: Colloque Géométrie Discrète en Imagerie: Fondements et Applications*. Université Louis Pasteur, pp. 108–134. – Strasbourg, septembre 1993.
- [Ris91] Risler (J.J.). – *Méthodes mathématiques pour la CAO*. – Paris, Masson, 1991, *Recherches en Mathématiques Appliquées*.
- [RK94] Rossignac (J.) et Kaul (A.). – AGRELS and BIBs: metamorphosis as a Bézier curve in the space of polyhedra. *In: Eurographics'94*, pp. C–179–C–184. – Oslo, septembre 1994.
- [Ros86] Rosenfeld (A.). – Axial representation of shape. *Computer Vision, Graphics, and Image Processing*, vol. 33, 1986, pp. 156–173.
- [Rou93] Roux (J.-C.). – Curves reconstruction. *In: Second international conference on curves and surfaces*, éd. par Laurent (P.-J.), Mehaute (A. Le) et Schumaker (L.). Association française d'approximation. – Chamonix, juin 1993.
- [Sch70] Schwartz (L.). – *Topologie générale et analyse fonctionnelle*. – Paris, Hermann, 1970, *Enseignement des sciences*, volume 2.
- [Sch89] Schmitt (M.). – Des algorithmes morphologiques à l'intelligence artificielle. – Thèse de doctorat de l'Ecole des Mines de Paris, février 1989.
- [Ser82] Serra (J.). – *Image analysis and mathematical morphology*. – Academic Press, vol. 1, 1982.
- [SES93] Séqueira (J.), Ebel (R.) et Schmitt (F.). – 3D-modeling of tree-like anatomical structures. *Computerized medical imaging and graphics*, vol. 17, n° 4, 1993.
- [SG92] Sederberg (T.W.) et Greenwood (E.). – A physically based approach to 2-D shape blending. *Computer Graphics (SIGGRAPH'92)*, vol. 26, n° 2, juillet 1992, pp. 25–34.
- [SGWM93] Sederberg (T.W.), Gao (P.), Wang (G.) et Mu (H.). – 2D shape blending: An intrinsic solution to the vertex path problem. *Computer Graphics (SIGGRAPH'93)*, vol. 27, n° 2, août 1993, pp. 15–18.

- [Sho87] Shoemake (K.). – Quaternion calculus and fast animation, computer animation: 3-D motion specification and control. *In: SIGGRAPH Course Notes*, pp. 101–121.
- [SK91] Shinagawa (Y.) et Kunii (T.L.). – Constructing a Reeb graph automatically from cross sections. *IEEE Computer Graphics and Applications*, vol. 11, n° 6, novembre 1991, pp. 44–51.
- [SKK91] Shinagawa (Y.), Kunii (T.L.) et Kergosien (Y. L.). – Surface coding based on Morse theory. *IEEE Computer Graphics and Applications*, vol. 11, n° 5, septembre 1991, pp. 66–78.
- [SMa88] Serra (J.), Matheron (G.) et al. – *Image analysis and mathematical morphology*. – Academic Press, vol. 2, 1988.
- [SP86] Sederberg (T.W.) et Parry (S.R.). – Free-form deformation of solid geometric models. *In: Computer Graphics (SIGGRAPH '86 Proceedings)*, pp. 151–160.
- [SR95] Shapira (M.) et Rappoport (A.). – Shape blending using the star-skeleton representation. *IEEE Computer Graphics and Applications*, vol. 15, n° 2, mars 1995, pp. 44–50.
- [ST92] Szeliski (R.) et Tonnesen (D.). – Surface modeling with oriented particle systems. *Computer Graphics (SIGGRAPH'92)*, vol. 26, n° 2, juillet 1992, pp. 185–194.
- [ST94] Sapiro (G.) et Tannenbaum (A.). – Area and length preserving geometric invariant scale-spaces. *In: Computer Vision*, éd. par Eklundh (Jan-Olof). ECCV'94, pp. 449–458. – Springer-Verlag.
- [Sut87] Sutcliffe (D.C.). – Contouring over rectangular and skewed rectangular grids - an introduction. *In: SIGGRAPH Course Notes*, pp. 69–91.
- [SV90] Subirana-Vilanova (J.B.). – Curved inertia frames and the skeleton sketch: finding salient frames of reference. *In: ICCV'90. IEEE*, pp. 702–708. – Osaka, Japan, 1990.
- [Ter94] Terraz (O.). – Programmation de métamorphoses d'objets surfaciques et volumiques. – Thèse de doctorat en sciences, Université Louis Pasteur, Strasbourg, septembre 1994.
- [TL95] Terraz (O.) et Lienhardt (P.). – A study of basix tools for simulating metamorphoses of subdivided 2D and 3D objects. Applications to the internal growing of

- wood and to the simulation of the growing of fishes. *In: Eurographics Workshop on Animation and Simulation*. – Maastrich, septembre 1995.
- [Uni93] Université Louis Pasteur. – *Colloque Géométrie Discrète en Imagerie: Fondements et Applications*. – Strasbourg, septembre 1993.
- [Wal76] Wall (C.T.C.). – Geometric properties of generic differentiable manifolds. *In: Geometry and Topology*. – Springer Verlag, 1976.
- [WH94] Witkin (A.) et Heckbert (P.). – Using particles to sample and control implicit surfaces. *In: Proceedings of SIGGRAPH '94 (Orlando, Florida, July 24–29, 1994)*, éd. par Glassner (Andrew). pp. 269–278. – ACM Press.
- [Wij91] Wijk (J.J. Van). – Rendering lines on curved surfaces. *In: Eurographics Workshop on Visualization in Scientific Computing*, éd. par Grave (M.), Lous (Y. Le) et Hewitt (W.T.). pp. 113–120. – Springer Verlag, 1994.

Table des figures

1.1	À gauche : interpolation au temps $1/2$ entre deux segments Γ_0 et Γ_1 du plan orientés dans le sens contraire. À droite : interpolation au temps $1/2$ entre deux arcs de cercle. La courbe interpolée est un arc de cardioïde.	12
1.2	Transport du vecteur $\gamma'(s)$ en $T_e X$	16
1.3	Le vecteur $v_t(s)$ est transporté en tout point de X	16
2.1	les angles α_i^n et θ_i^n (les indices n sont supprimés sur la figure).	36
2.2	Influence d'une torsion supplémentaire.	40
2.3	Influence du contrôle de la vitesse d'interpolation. Les courbes interpolées sont visualisées aux temps $t = 0, 0.1, 0.2, 0.35, 0.5, 0.75, 1$	41
2.4	Redressement automatique d'une courbe	42
2.5	Interpolation par la méthode A.	44
2.6	Interpolation par la méthode Q.	45
1.1	La réunion d'un carré C unitaire et de son translaté $C_{1+1/n}$, le long d'une de ces arêtes, converge dans \mathcal{F} vers un rectangle lorsque la translation de longueur $1+1/n$ tend vers 1. Cependant la limite supérieure du squelette de ce $C \cap C_{1+1/n}$ n'est pas contenue dans le squelette du rectangle, et la limite inférieure ne contient pas le squelette du rectangle.	53
1.2	Squelette d'un parallélépipède.	55
1.3	(a) Un tore, (b) ses sections et (c) son diagramme de Reeb.	56
2.1	La fonction de Morse hauteur sur la sphère se transporte sur les formes qui lui sont difféomorphes.	59
2.2	Les barycentres des lignes de niveau fournissent une courbe qui présente un point de rebroussement à l'endroit où la calotte est enfoncée.	60
2.3	Les barycentres des composantes connexes des courbes de niveau ne forment pas un ensemble connexe.	60
2.4	Un objet qui n'est monotone par rapport à aucune direction.	61
2.5	r est le rayon d'injectivité du point s	62

3.1	Les géodésiques se déplient en segment de droite.	64
3.2	Propagation d'un intervalle d'optimalité.	65
3.3	Les faces sont complétés par toutes arêtes possibles entre leurs sommets.	66
3.4	Un découpage en parties cylindriques. Les points noirs représentent des ensembles critiques qui doivent être ôtés.	67
3.5	Les voisins du sommet s	69
3.6	Les lignes de champ sur un triangle. La configuration représentée a toujours lieu : une arête a une orientation opposée aux deux autres, plus généralement il n'existe aucun cycle d'arêtes d'orientations compatibles sur le polyèdre.	70
3.7	L'indice du sommet s est nul.	70
3.8	La ligne de niveau traverse les arêtes tracées en trait gras.	71
3.9	Les trois possibilités de raccord des lignes de niveau dans un rectangle dégénéré.	72
3.10	À gauche : interpolation linéaire des courbes de niveau. À droite : l'interpolation spline peut générer des incohérences.	72
3.11	À gauche : la configuration générale. À droite : le cas orienté.	74
3.12	Deux exemples de modifications de contour au voisinage d'un sommet s . Les "+" (resp. "-") symbolisent des sommets de couleurs supérieures (resp. inférieures) à $c(s)$. La courbe en trait pointillé (resp. plein) représente la ligne de niveau inférieure (resp. supérieure) à $c(s)$	75
3.13	À droite : le sommet σ_2 de couleur $c(s_i)$ d'indice nul. À gauche : le graphe des arêtes correspondantes. \mathcal{C}_i contient la chaîne (a_1, a_2, a_3, a_4) qui est remplacée par $(a_1, a_5, a_6, a_7, a_8, a_4)$ dans \mathcal{C}_{i+1}	76
3.14	Une bande cylindrique entre deux niveaux $c(s_i)$ et $c(s_{i+1})$	78
3.15	L'arête de sortie (resp. d'entrée) a_s (resp. a_e) est alors la première arête rencontrée transverse à c_0	78
3.16	Les voisins de s_i ont une couleur inférieure à $c(s_i)$	81
3.17	\mathcal{C}^j entoure \mathcal{C}^i qui entoure σ et \mathcal{C}^i entoure \mathcal{C}^j qui entoure s !	82
3.18	Au dessus : la ligne de niveau légèrement inférieur à $c(s_i)$. En dessous : en trait plein, la ligne de niveau légèrement supérieur à $c(s_i)$. L'indice du sommet vaut -1 et il y a 1 création de cycle.	83
3.19	Les arêtes en pointillés se substituent aux arêtes en trait plein et gras et le cycle est coupé à cet endroit.	84
3.20	À gauche : la structure d'arête ailée pour un polyèdre orientable. À droite : le cas d'une triangulation.	85
3.21	Un cône tronqué par un plan oblique et tel que $d(s_i) < d(s_{i+1})$	87
3.22	Trois courbes de niveau. Le petit triangle indique le point source.	88

3.23	Le long de la géodésique d'un maximum local s_M on rencontre une face qui contient une arête du contour cherché.	89
3.24	L'orientation de a et de \mathcal{P} permet de distinguer une face f_s et f_e	89
3.25	Un cygne, son axe et 20 courbes de niveau.	91
3.26	À gauche : les flèches noires et blanches désignent les deux points sources utilisés pour le calcul de l'axe. Au milieu : les deux axes correspondant sont représentés sur la même figure. À droite : un détail de la figure du milieu.	92
3.27	Deux calculs d'axe.	92
3.28	À gauche : une vue de face de l'objet et l'axe calculé en trait épais. Au milieu : une vue latérale de l'objet. À droite : la même vue de l'axe.	93
3.29	Autres exemples de calcul d'axe.	93
3.30	À gauche : la condition d'étoilement se traduit par $\langle \gamma(s)p, \mathbf{n}_p \rangle > 0$. À droite : la condition de régularité du paramétrage se traduit par la position de $\mathcal{C}(s)$ par rapport à la binormale translatée au centre de courbure.	94
1.1	Le graphe associé à une image. Une seule composante de la couleur est représentée.	104
1.2	Les traits caractéristiques sont constitués de deux segments. Les déformations W_0 et W_1 ne sont pas nécessairement inverses l'une de l'autre.	105
1.3	Différents types de faces dans la somme de Minkowski de deux polyèdres A et B (figure extraite de [KR91]).	108
1.4	La mise en correspondance de deux contours P et Q peut être représentée par un graphe torique. Chaque point de ce graphe représente une arête de liaison entre les deux contours.	109
1.5	Deux décompositions compatibles de polygones. Les points \bullet représentent les sommets des polygones, les points carrés et noirs représentent les centres d'étoilement des sous-parties et les points \square représentent les milieux des arêtes d'adjacences entre deux sous-parties. Le point 8 est la racine de l'arbre associé aux sous-parties (figure extraite de [SR95].)	111
1.6	Subdivision récursive d'un tétraèdre et d'un cube. Seule la moitié de l'arbre de subdivision est montrée (figure extraite de [Par92]).	113
1.7	Principe (en deux dimensions) de l'interpolation entre un triangle et un carré.	114
2.1	Un axe permet de décomposer un objet en trois parties.	118
2.2	Le point p est projeté orthogonalement sur γ	119
2.3	Le point p peut être projeté comme sur les deux figures, mais seul l'attachement de la figure de gauche peut aboutir à un attachement global de l'objet.	120
2.4	La fonction d'attachement est discontinue au niveau des points noirs. Une déformation de l'axe produit un objet discontinu.	120

2.5	Le dièdre D_i	121
2.6	Report du système de coordonnées du plan Π_i dans le plan Π_{i+1}	122
2.7	Le point p doit être attaché différemment selon l'objet auquel il appartient.	123
2.8	L'objet gris peut être attaché aussi bien sur la portion d'axe comprise entre s_1 et s_2 qu'entre celle comprise entre s_3 et s_4	123
2.9	Le sommet p est attaché en progressant le long de Γ à partir du point Γ_i	124
2.10	Les parties grisées de l'objet en trait épais sont attachées aux extrémités de l'axe γ	125
2.11	La face $F = ABCD$ est projetée dans l'espace des paramètres. Les échantillons intérieurs sont calculés par balayage à l'aide des plans de coupe.	129
2.12	On détermine les échantillons dans l'intersection ab de la face F avec le plan de coupe Π_i	129
2.13	On détermine les échantillons dans l'intersection ab de la face F avec le quart de plan de coupe associé à l'angle θ_i	130
2.14	Un objet ouvert au niveau de la calotte supérieure. Les échantillons trouvés dans les cartes correspondent à la partie grise des cartes.	130
2.15	Les échantillons de paramètres $\phi_i = \frac{\pi}{2}$ sont identifiés.	131
2.16	Les secteurs angulaires associés aux faces F_1 et F_2 sont figurés par des zones hachurées.	132
2.17	À gauche : un cube et son axe. Au milieu : le cube échantillonné avec la procédure précédente. À droite : le maillage recalé.	133
2.18	À gauche : l'angle α entre les normales des deux faces incidentes à une arête permet de déterminer si cette arête est vive. À droite : l'angle α maximal entre les normales des faces incidente à un sommet permet de le sélectionner comme sommet conique.	133
2.19	La zone de déplacement associée à un échantillon $p_{i,j}$ dans l'espace des paramètres.	134
2.20	Le point $q_{i,j}$ est l'intersection de la droite support de \mathcal{A} et de la droite déterminée par les deux plans orthogonaux passant par l'extrémité de l'axe et contenant respectivement l'arête \mathcal{A} et le point $p_{i,j}$	135
2.21	La courbe noire représente la projection d'une arête dans l'espace des paramètres. Les points blancs sont les échantillons à déplacer et leur zone de déplacement est représentée par des rectangles gris. Les droites horizontales en pointillés indiquent les plans de coupe intermédiaires, tandis que les points noirs sont les intersections de ces plans avec l'arête considérée (et les extrémités de l'arête). Sur la figure de droite on montre le maillage déformé et les diagonales ajoutées sont indiquées en trait épais.	136
2.22	Un sommet $p_{i,j}$ du maillage recalé peut avoir huit voisins si des diagonales ont été ajoutées dans chacune des mailles incidentes.	137

2.23	Le surmaillage est obtenu par superposition des grilles de chaque objet recalé. . .	138
2.24	Paramétrisation de la partie cylindrique et des calottes.	140
2.25	La figure représente une coupe des objets par un plan vertical passant par l'axe. Les points en correspondance sur le bord doivent avoir des tangentes parallèles pour obtenir une interpolation régulière.	140
2.26	Transformation entre deux cônes tronqués. Les axes sont identiques et visualisés en traits épais. Les objets sont représentés aux instants 0, 1/2 et 1. Le grand rectangle indique le niveau de raccord entre la partie cylindrique et les calottes. .	141
2.27	Transformation entre deux cônes tronqués. Les angles entre les plans tangents aux deux cônes sont plus importants que sur la figure 2.26.	141
2.28	Le même axe est utilisé pour échantillonner les deux objets clefs.	142
2.29	L'axe initial et l'axe final sont échangés entre la transformation de la rangée du haut et celle de la rangée du bas.	142
2.30	Un rendu simple de la figure 2.29.	143
2.31	Les formes de la rangée du milieu sont interpolées entre les formes du haut et du bas.	143
2.32	Transformation d'une calotte sphérique en sphère.	144
2.33	Un rendu de la figure 2.32.	144
2.34	Transformation d'une faucille en poisson.	145
2.35	Deux métamorphoses extraites du film Metadata.	146
2.36	Deux métamorphoses extraites du film Metadata.	146
C.1	Une carte autour d'une rotation R_0	156
E.1	Les fermés F_1 et F_2 sont dans l'ouvert $\mathcal{F}_K^{G_1, G_2}$ défini par les ouverts G_1 et G_2 et le compact K	164
F.1	H est un champ tangent aux branches d'hyperboles $x^2 - y^2 = k^2$	168
F.2	Changement de coordonnées projectives.	169
F.3	Le champ se prolonge continûment sur les bords de Δ	170
F.4	À gauche : le profil de h . À droite : le graphe de h_2	170
F.5	Le champ H_2	171