



**HAL**  
open science

# Modélisation Hiérarchique de Surfaces à partir de Maillages Polyédriques et Applications

Alex Yvart

► **To cite this version:**

Alex Yvart. Modélisation Hiérarchique de Surfaces à partir de Maillages Polyédriques et Applications. domain\_stic.comm. Institut National Polytechnique de Grenoble - INPG, 2004. Français. NNT : . tel-00009003

**HAL Id: tel-00009003**

**<https://theses.hal.science/tel-00009003>**

Submitted on 12 Apr 2005

**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.

INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE

**THESE**

pour obtenir le grade de

**DOCTEUR DE L'INPG**

*Spécialité* : « **Imagerie, Vision et Robotique** »

préparée au Laboratoire de Modélisation et Calcul (LMC / IMAG)

dans le cadre de l'**Ecole Doctorale**

« **Mathématiques, Sciences et Technologies de l'Information** »

présentée et soutenue publiquement par

Alex YVART

le 13 Décembre 2004

**Modélisation Hiérarchique de Surfaces à partir de  
Maillages Polyédriques et Applications**

---

*Directrice de thèse* : Stefanie HAHMANN

*Co-Directeur* : Georges-Pierre BONNEAU

---

**JURY**

M. Marc DANIEL	Professeur - Université de la Méditerranée	Rapporteur
M. Marc NEVEU	Professeur - Université de Bourgogne	Rapporteur
M <sup>me</sup> Marie-Paule CANI	Professeur - INP de Grenoble	Examinatrice
M <sup>me</sup> Géraldine MORIN	Maître de Conférences - INP de Toulouse	Examinatrice
M. Bernard LACOLLE	Professeur - Université Joseph Fourier	Examinateur
M <sup>me</sup> Stefanie HAHMANN	Maître de Conférences - INP de Grenoble	Directrice de thèse
M. Georges-Pierre BONNEAU	Professeur - Université Joseph Fourier	Co-directeur de thèse



## Modélisation Hiérarchique de Surfaces à partir de Maillages Polyédriques et Applications.

La modélisation géométrique est incontournable dans la création et l'élaboration d'un produit. De même, l'infographie est couramment employée en effets spéciaux ou réalité virtuelle. Pourtant, chaque domaine utilise ses propres outils pour modéliser des surfaces.

Pour répondre aux contraintes de ces deux types d'utilisation, nous proposons une nouvelle modélisation de surfaces polynomiales, interpolantes, globalement lisses (de plan tangent continu) et hiérarchique. Une surface lisse initiale est d'abord construite sur un maillage triangulaire, ce qui permet de traiter toutes les topologies. Des détails sont ensuite progressivement ajoutés par niveaux. Pour ce faire, chaque face est subdivisée en quatre sous-faces par insertion d'un sommet en milieu d'arête. Ce raffinement n'induit pas de modifications significatives sur la surface. Chaque détail est défini localement par rapport au précédent grâce à l'utilisation de repères locaux. Cette hiérarchie permet à l'ensemble des détails fins de suivre continûment le déplacement lors d'une édition de la surface.

Deux applications sont ensuite proposées dans cet ouvrage : Tout d'abord, un modéleur 3D respectant la démarche créative des artistes. Celui-ci repose sur le calcul d'une forme globale progressivement affinée pour obtenir l'objet désiré, et offre la possibilité d'éditer les objets de manière intuitive en modifiant très peu de paramètres. Enfin, un outil de reconstruction est présenté pour modéliser des objets existants grâce à notre nouvelle représentation hiérarchique.

**Mots-clés :** Surface hiérarchique, niveaux de détail, interpolation polynomiale, modélisation géométrique, patch de Bézier, subdivision, raffinement.



# Remerciements

Je tiens à remercier Stefanie HAHMANN et Georges-Pierre BONNEAU pour avoir encadré cette thèse, m'avoir intégré au monde de la recherche et beaucoup aidé quand le temps manquait.

Je remercie M. Marc NEVEU et M. Marc DANIEL d'avoir accepté d'être les rapporteurs de cette thèse. Mes remerciements vont également aux membres du jury : M<sup>me</sup> Marie-Paule CANI, M<sup>me</sup> Géraldine MORIN et M. Bernard LACOLLE.

Ce mémoire est le résultat de trois années d'étude. Trois années pendant lesquelles beaucoup de monde m'a entouré et aidé. Si je devais les remercier tous nominativement, il me faudrait de nombreuses pages. Je vais donc le faire collectivement. A vous tous qui m'avez soutenu, motivé mais aussi distrait, je dis un grand **merci**. Je pense notamment, mais pas exclusivement, à Claudine MEYRIEUX, grâce à qui les démarches administratives les plus lourdes deviennent simples et Dominique DUVAL qui m'a fait découvrir la richesse de l'enseignement. Je n'oublie pas l'équipe MGA du LMC qui m'a accueilli, le LMC tout entier, ainsi que l'équipe Evasion et tout le laboratoire Gravir. En particulier, les doctorants ont réussi à instaurer une bonne ambiance, entre autres pendant les pauses-repas. Je les en remercie. Les permanents m'ont aussi beaucoup apporté, merci à eux.

Un remerciement particulier va à Basile, qui a eu la patience de me supporter tous les jours et dans tous les moments, pas tous bons. Nous avons refait le monde de nombreuses fois, mais il m'a aussi donné envie de reprendre ma clarinette par sa passion de la musique.

Un grand merci va également à mes parents. Ils m'ont permis de mener les études qui m'ont amené à ce mémoire, et toujours soutenu et encouragé quels que soient les événements. Merci.

Enfin, il me faut parler de celle qui m'a le plus soutenu et entouré. Elle a su me motiver même dans les pires moments de découragement et m'a permis de travailler sans arrêt les jours de motivation extrême. C'est grâce à elle que ce mémoire est né. Pour tout ça et beaucoup plus encore, je lui dis merci d'illuminer ma vie.



# Sommaire

<b>Introduction</b>	<b>1</b>
Problématique . . . . .	2
Contributions . . . . .	3
Organisation du mémoire . . . . .	3
<b>1 Modélisation de surfaces</b>	<b>5</b>
Introduction . . . . .	6
1.1 Surfaces polynomiales . . . . .	6
1.1.1 Présentation . . . . .	6
1.1.2 Produit Tensoriel . . . . .	7
1.1.3 Maillages triangulaires . . . . .	11
1.1.4 Conclusion . . . . .	22
1.2 Surfaces de subdivision . . . . .	23
1.2.1 Présentation . . . . .	23
1.2.2 Conclusion . . . . .	27
1.3 Surfaces par ondelettes, Analyse multirésolution . . . . .	27
1.3.1 Présentation . . . . .	27
1.3.2 Conclusion . . . . .	29
1.4 Maillages hiérarchiques . . . . .	29
1.5 Discussion . . . . .	30
Conclusion . . . . .	31
<b>2 Interpolation hiérarchique</b>	<b>33</b>
Introduction . . . . .	34
2.1 Raffinement paramétrique . . . . .	35
2.2 Subdivision et structure de données . . . . .	39
2.3 Interpolation polynomiale . . . . .	42
2.3.1 Continuité du plan tangent . . . . .	43



2.3.2	Continuité du plan tangent aux sommets . . . . .	43
2.3.3	Consistance aux sommets . . . . .	46
2.3.4	Courbes frontières . . . . .	48
2.3.5	Rubans de tangence . . . . .	50
2.3.6	Points libres intérieurs . . . . .	51
2.3.7	Paramètres libres . . . . .	52
2.3.8	Modifications en vue de la hiérarchisation . . . . .	54
2.3.9	Comparaison avec la méthode Butterfly . . . . .	54
2.4	Raffinement local . . . . .	55
2.4.1	Invariance le long des courbes frontières . . . . .	59
2.4.2	Erreur le long des courbes intérieures . . . . .	70
2.5	Raffinement et édition hiérarchique . . . . .	77
2.5.1	Aspects topologiques . . . . .	77
2.5.2	Aspects géométriques . . . . .	81
2.6	Résultats . . . . .	81
	Conclusion . . . . .	85
<b>3</b>	<b>Applications</b>	<b>87</b>
	Introduction . . . . .	88
3.1	Modeleur 3D . . . . .	89
3.2	Reconstruction de surface . . . . .	92
3.2.1	Reconstruction de surfaces linéaires . . . . .	95
3.2.2	Reconstruction de surfaces lisses . . . . .	99
3.2.3	Conclusion . . . . .	117
	Conclusion . . . . .	123
	<b>Conclusion</b>	<b>125</b>
	<b>Annexes</b>	<b>131</b>
<b>A</b>	<b>Notions de CAGD</b>	<b>131</b>
A.1	Contuinuité géométrique . . . . .	131
A.1.1	Méthode de Farin . . . . .	132
A.1.2	Méthode de Chiyokura-Kimura . . . . .	133
<b>B</b>	<b>Format de fichier hiérarchique</b>	<b>135</b>
<b>C</b>	<b>Algorithme de subdivision de Goldman</b>	<b>139</b>

**Bibliographie**

**143**



# Table des figures

1.1.1 Configurations non produit tensoriel . . . . .	8
1.1.2 Modélisation par courbes de découpe (trim curves) - Images extraites du livre « OpenGL Optimizer Programmer's Guide : An Open API for Large-Model Visualization » . . . . .	9
1.1.3 Raffinement le long de la diagonale : cas le pire. Sans les overlays, tout devrait être subdivisé - Images extraites de la page web de David Forsey	9
1.1.4 Définition d'un overlay : édition avant et après ajout d'un overlay - Images extraites du site web de Wayne Olive . . . . .	10
1.1.5 Déformation globale avec conservation des détails - Images extraites du site web de David Forsey . . . . .	10
1.1.6 (a) Exemple de maillage. Les 9 points indiqués définissent quatre patches triangulaires de Bézier (b). (c) Un raffinement local et sa surface (d). (e) Raffinement global et sa surface (f) - Images extraites de [Gonzalez-Ochoa et Peters, 1999]. . . . .	12
1.1.7 Exemple de changement de topologie. Deux protubérances (a) fusionnent en une anse (b). Un trou est alors percé (c) (d) - Images extraites de [Gonzalez-Ochoa et Peters, 1999]. . . . .	13
1.1.8 Problème de consistance en un sommet . . . . .	15
1.1.9 Subdivision de Clough-Tocher . . . . .	16
1.1.10 Face de Bézier cubique . . . . .	17
1.1.11 Patch de Gregory . . . . .	18
1.1.12 Paramétrisation autour d'un sommet. . . . .	20
1.1.13 Découpage de l'espace des paramètres en quatre. . . . .	21
1.1.14 Courbes de bord autour d'un sommet. Au milieu, courbes issues de [Hahmann et Bonneau, 2003]. Sur la droite, les courbes sont issues de [Bonneau et Hahmann, 2000]. . . . .	22
1.2.1 Surfaces de Subdivision : une étape de raffinement . . . . .	23

1.2.2 Surfaces de Subdivision : (a) maillage de contrôle, (b) après une étape de subdivision, (c) après deux étapes de subdivision, (d) la surface de subdivision - Image extraite de [DeRose <i>et al.</i> , 1998] . . . . .	24
1.2.3 Masque du schéma butterfly - Image extraite de [Hubeli et Gross, 2000]	25
1.2.4 Exemple de calcul de surface à l'aide du schéma butterfly - Image extraite de [Hubeli et Gross, 2000] . . . . .	26
1.2.5 Masque du schéma butterfly modifié - image extraite de [Zorin <i>et al.</i> , 1996]	26
1.2.6 Masque du schéma butterfly modifié - image extraite de [Zorin <i>et al.</i> , 1996]	27
1.3.1 Décomposition d'un maillage en ondelettes : calcul d'un maillage + des coefficients de détail - Image extraite de [Lounsbery <i>et al.</i> , 1997] .	28
1.4.1 Rendu dépendant du point de vue avec les maillages progressifs - Image extraite de [Hoppe, 1996] . . . . .	29
2.1.1 Ajout de détail avant et après raffinement : la taille du détail dépend du niveau de raffinement. . . . .	36
2.1.2 Principe des niveaux de détails : A partir d'une surface initiale, les détails sont ajoutés en ordre décroissant de leur taille. . . . .	37
2.1.3 Le raffinement et l'édition spline conservent la continuité du plan tangent, alors que la subdivision et l'édition de Bézier peuvent détruire cette continuité. Rangée supérieure : courbes de Hermite - raffinement local - édition local qui maintient la continuité $C^1$ . Rangée inférieure : courbes de Bézier - subdivision de Bézier - déformation locale détruisant la continuité $G^1$ . . . . .	38
2.2.1 Deux patches adjacents sont subdivisés en quatre sous-patches en insérant un nouveau sommet au milieu de leur arête commune. Ce sommet est un sommet <i>éditable</i> . Sa position peu être modifiée, générant une modification locale de la surface tout en conservant la continuité du plan tangent de la surface. . . . .	40
2.2.2 Subdivision locale d'une surface : deux types de sommets sont créés. .	40
2.2.3 La subdivision en quatre permet d'insérer des détails aussi facilement le long des arêtes ou à l'intérieur des faces. . . . .	41
2.3.1 Paramétrisation de l'interpolant. Gauche : Espace des paramètres ; droite : macro-patches avec les directions de tangence. . . . .	42
2.3.2 Notations indexées par le numéro de face . . . . .	44
2.3.3 Calcul de l'interpolant triangulaire en 4 étapes. . . . .	45
2.3.4 Notations utilisée le long des courbes de bord. . . . .	47

2.3.5 Points de contrôle intervenants dans la continuité intérieure. A droite, règle de parallélogramme. . . . .	51
2.3.6 Points de contrôle d'un macro-patch composé de quatre triangles de Bézier quintiques. Les points de contrôle symbolisés par des cercles sont traités dans l'étape (S1). Les points de contrôle carrés sont calculés en étapes (S2) et (S3). L'étape (S4) implique les points de contrôle représentés par des triangles. . . . .	52
2.3.7 Influence de la constante multiplicative des vecteurs dérivées premières sur la surface - image extraite de [Hahmann et Bonneau, 2003].	53
2.3.8 Comparaison de la méthode issue de [Hahmann et Bonneau, 2003] et du butterfly. . . . .	54
2.4.1 Raffinement local. . . . .	56
2.4.2 Raffinement local. Gauche : la continuité tangent avec la surface environnante est assurée par une subdivision de Bézier. Milieu et droite : La méthode d'interpolation de la section 2.3 est appliquée localement pour calculer la surface raffinée. . . . .	57
2.4.3 Notations utilisée le long des courbes de bord. . . . .	59
2.4.4 Notations utilisée pour l'intérieur. . . . .	70
2.4.5 Erreur introduite par le raffinement. Les cercles représentent les points exacts, les carrés des points introduisant une erreur. . . . .	76
2.4.6 Observation de l'erreur après une étape de raffinement. . . . .	78
2.4.7 Observation de l'erreur dans un cas défavorable. . . . .	79
2.5.1 5 types d'arêtes se cotoient dans la structure de données hiérarchique	80
2.5.2 Après les raffinements successifs des sommets $A$ et $C$ , le sommet $B$ devient éditable. . . . .	80
2.5.3 Edition Hiérarchique à l'aide d'un repère local. . . . .	81
2.6.1 Surface lisse à partir d'un maillage de genre 1 : raffinements, éditions locales et éditions globales. . . . .	83
2.6.2 Modèle d'une tête de chien. . . . .	84
2.6.3 Edition globale : Un seul sommet est modifié. . . . .	85
3.1.1 Modeleur 3D . . . . .	89
3.1.2 Exemples d'opérations réalisées par notre modeleur. . . . .	91
3.2.1 Reconstruction hiérarchique linéaire. . . . .	96
3.2.2 Reconstruction du grand canyon. . . . .	97
3.2.3 Déplacement du grand canyon. . . . .	98
3.2.4 Données de la tête chinoise avant lissage. . . . .	99

3.2.5	Données de la tête chinoise après lissage. . . . .	100
3.2.6	Reconstruction de la tête chinoise lissée. . . . .	101
3.2.7	Reconstruction de la tête chinoise lissée. . . . .	102
3.2.8	Création du graphe à partir du maillage. . . . .	105
3.2.9	Chemins de longueur minimale. . . . .	105
3.2.10	Courbes tendues. . . . .	106
3.2.11	Lissage de la polyligne. Les faces sont dépliées, la polyligne est remplacée par un segment, puis on replie. Localement, la longueur de la polyligne est minimisée. . . . .	107
3.2.12	Paramétrisation autour d'un sommet et définition des angles - Image extraite de [Desbrun <i>et al.</i> , 2002]. . . . .	108
3.2.13	Echantillon uniforme. . . . .	109
3.2.14	Lissage de l'échantillon par paramétrisation couplée. Chaque courbe diagonale est remplacée par l'image de la diagonale du carré dans la paramétrisation des deux patches voisins. . . . .	110
3.2.15	Résultats après lissage. . . . .	110
3.2.16	Paramétrisation du lapin de Stanford. . . . .	112
3.2.17	Reconstruction du lapin de Stanford. . . . .	118
3.2.18	Reconstruction du buste de Max Planck. . . . .	119
3.2.19	Reconstructions locales. . . . .	120
3.2.20	Répartition de l'erreur du lapin : Nombre de points compris entre deux seuils d'erreur. . . . .	120
3.2.21	Répartition de l'erreur pour Max Planck : Nombre de points compris entre deux seuils d'erreur. . . . .	121
A.1.1	Patch $Q$ et $R$ partageant une courbe de bord. . . . .	131
A.1.2	Points de contrôle des patches $Q$ et $R$ . . . . .	132

# Introduction

## Sommaire

---

<b>Problématique</b> . . . . .	<b>2</b>
<b>Contributions</b> . . . . .	<b>3</b>
<b>Organisation du mémoire</b> . . . . .	<b>3</b>

---

La conception et la fabrication assistées par ordinateur (C.F.A.O.) regroupent l'ensemble des technologies mises en œuvre pour élaborer un nouveau produit. La première étape de création est bien sûr sa modélisation en trois dimensions à l'aide de courbes et de surfaces. Historiquement, les industries automobiles, navales et aéronautiques ont été les premières à utiliser ces outils. C'est pourquoi les premiers travaux en sont issus : P. Bézier chez Renault et P.F. De Casteljau chez Citroën ont introduit simultanément les courbes et surfaces de Bézier (seul P. Bézier a publié ses travaux), Coons chez General Motors et Jordan chez Ford ont développé les facettes de Coons interpolant un réseau de courbes. L'informatique s'est depuis généralisée, et toutes les industries utilisent la modélisation géométrique devenue incontournable dans le processus de création. Une nouvelle application des surfaces est par ailleurs apparue avec l'informatique graphique, que ce soit pour les effets spéciaux des grandes réalisations cinématographiques ou pour les simulateurs et autres jeux vidéo.

La puissance des ordinateurs augmente rapidement (double tous les dix-huit mois d'après la loi de Moore) et donne accès à des modèles de surfaces qui étaient jusqu'alors hors de portée. On constate une évolution des modèles utilisés ainsi qu'une augmentation de l'automatisation. L'utilisateur a de moins en moins besoin d'intervenir grâce aux performances croissantes des machines et des modèles. Les surfaces les plus utilisées restent les B-splines. Ce sont des surfaces de type produit tensoriel définies par une carte polynomiale d'un domaine planaire subdivisé en une grille régulière. Un objet est représenté par un ensemble de surfaces B-splines. Cela amène la question du raccordement de ces surfaces ( $C^k$  continuité). Par ailleurs, tous les objets ne peuvent pas être représentés sur une grille régulière. La topologie des



surfaces considérées s'en trouve réduite. De plus, l'édition de ces surfaces est délicate en dehors des lignes de la grille régulière.

D'autres types de surfaces les supplantent progressivement dans certaines applications spécifiques. On peut ainsi citer les surfaces de subdivision en réalité virtuelle qui présentent l'avantage de pouvoir modéliser tous les types d'objets (topologie quelconque, avec ou sans bord, genre quelconque) avec une grande simplicité algorithmique. Ces surfaces souffrent en revanche d'une absence de définition analytique. Certains calculs et évaluations sur la surface ne peuvent pas être effectués. Les surfaces à base d'ondelettes sont le fruit de l'analyse multirésolution. Elles permettent une représentation à niveau de détails : la précision de l'objet peut être modifiée en fonction de l'utilisation qui en est faite. Utilisées pour la compression et la transmission d'objets virtuels, elles reposent sur une théorie solide. Néanmoins, la reconstruction d'objets réels s'avère délicate. Pour la modélisation à grande précision d'objets réels, les maillages sont directement manipulés. Ils peuvent aussi être hiérarchisés pour exploiter les avantages d'une représentation avec des niveaux de détails. Leur taille prohibitive rend toutefois difficile leur exploitation.

## Problématique

Les applications visées dans le cadre de cette thèse sont la C.F.A.O. et la réalité virtuelle. Nous nous sommes donc efforcés de regrouper au sein d'un seul modèle les exigences de chacune de ces applications pour unifier ces deux domaines. Dans le cadre de la C.F.A.O., les surfaces utilisées sont définies par une carte polynomiale afin de disposer d'un évaluateur pour pouvoir facilement effectuer des calculs. La jonction de ces surfaces polynomiales implique une notion de continuité. Elles doivent se raccorder de manière visuellement lisse, ce qui peut se traduire mathématiquement par une continuité du plan tangent. Tous les types de surfaces doivent pouvoir être modélisés, quels que soient leur topologie, leur genre et l'existence de bord. Pour cette raison, les surfaces de type produit tensoriel sont abandonnées au profit de surfaces définies sur des maillages triangulaires.

Il est par ailleurs important de fournir aux utilisateurs des méthodes simples et intuitives d'édition. Une surface interpolante (passant par les sommets de la triangulation) est ainsi préférée. L'utilisateur manipule directement des points de la surface et non des points de contrôle. De plus, un principe de niveau de détails permet d'ajouter des détails localement comme le désire l'utilisateur. La surface est plus précise à certains endroits détaillés. Il est alors possible de construire un objet

à travers plusieurs couches : une première donne la forme globale puis de plus en plus de détails sont ajoutés à l'aide de couches successives. De plus, un rendu à différents niveaux de détails s'avère très pratique en infographie pour économiser des ressources graphiques afin d'obtenir un affichage interactif.

L'objectif de cette thèse est donc de développer un modèle de surface polynomiale, interpolante, globalement lisse dans le sens continuité du plan tangent et hiérarchique (à niveau de détails).

## Contributions

Une nouvelle représentation de surface a été développée dans le cadre de cette thèse. Fondée sur carte polynomiale d'une triangulation, elle permet de modéliser une surface de n'importe quel type topologique. Cette méthode est hiérarchique dans le sens où plusieurs niveaux de détails sont utilisés pour représenter un objet. Un maillage triangulaire grossier donne la topologie de l'objet. Sur ce maillage, une surface lisse de base est calculée. Elle est ensuite raffinée localement pour ajouter des détails. Cette structure hiérarchique facilite aussi l'édition : une édition globale conservera les détails locaux. La modification de surface est par là rendue intuitive et pratique si le maillage grossier est adapté.

A l'aide de cette méthode, des objets réels sont reconstruits grâce à une technique de « fitting ». Le maillage grossier ainsi que les différents niveaux de détails sont optimisés pour approcher les données.

## Organisation du mémoire

Ce document est constitué de trois grandes parties.

Le chapitre 1 est consacré à un état de l'art des modélisations de surfaces rappelant les différentes approches envisagées ainsi que leurs avantages et défauts. Les méthodes polynomiales sont envisagées au chapitre 1.1 à travers les surfaces de type produit tensoriel (chapitre 1.1.2) ainsi que les surfaces définies sur un maillage triangulaire (chapitre 1.1.3). Les surfaces de subdivision sont discutées au chapitre 1.2 avant de passer aux surfaces par ondelettes et l'analyse multirésolution au chapitre 1.3. Les méthodes à base de maillages progressifs sont traitées au chapitre 1.4.

Le chapitre 2 présente la méthode hiérarchique que nous avons mise au point. Tout d'abord une définition du raffinement paramétrique est donnée (chapitre 2.1). Ensuite la subdivision est introduite (chapitre 2.2). Le choix de l'interpolant poly-

nomial est discuté chapitre 2.3 avec les modifications nécessaires en vu de sa hiérarchisation. La technique de raffinement local est détaillée chapitre 2.4. L'édition hiérarchique (chapitre 2.5.2) devient alors possible. Les résultats et exemples sont alors donnés chapitre 2.6.

Le chapitre 3 regroupe des applications de notre méthode. Ainsi nous avons développé un modeleur 3D (chapitre 3.1) qui permet à un designer d'utiliser notre représentation pour créer des objets. Dans le cas d'objets réels, ils peuvent être modélisés grâce à l'application de reconstruction décrite au chapitre 3.2.

# Chapitre 1

## Modélisation de surfaces

### Sommaire

---

<b>Introduction</b> . . . . .	<b>6</b>
<b>1.1 Surfaces polynomiales</b> . . . . .	<b>6</b>
1.1.1 Présentation . . . . .	6
1.1.2 Produit Tensoriel . . . . .	7
H-Splines . . . . .	9
Localized-Hierarchy Surface Splines (LeSS) . . . . .	11
1.1.3 Maillages triangulaires . . . . .	11
Problème de la consistance aux sommets . . . . .	14
Stratégie par subdivision du domaine . . . . .	16
Méthode de Piper . . . . .	16
Méthode de Shirman Sequin . . . . .	17
Méthode de Jensen . . . . .	18
Stratégie par Combinaison convexe . . . . .	18
Patches triangulaires de Gregory . . . . .	18
Stratégie par facettes dégénérées . . . . .	19
Stratégie par Courbes frontières . . . . .	19
Méthode de Loop . . . . .	20
Méthode de Hahmann-Bonneau . . . . .	21
Méthode de Peters . . . . .	22
1.1.4 Conclusion . . . . .	22
<b>1.2 Surfaces de subdivision</b> . . . . .	<b>23</b>
1.2.1 Présentation . . . . .	23
1.2.2 Conclusion . . . . .	27

---

<b>1.3 Surfaces par ondelettes, Analyse multirésolution . . . .</b>	<b>27</b>
1.3.1 Présentation . . . . .	27
1.3.2 Conclusion . . . . .	29
<b>1.4 Maillages hiérarchiques . . . . .</b>	<b>29</b>
<b>1.5 Discussion . . . . .</b>	<b>30</b>
<b>Conclusion . . . . .</b>	<b>31</b>

---

## Introduction

La modélisation de surfaces est devenue un outil incontournable dans l'industrie moderne. En effet, elle permet de représenter et d'étudier à l'aide d'un ensemble d'équations des surfaces constituant des objets. Dans un premier temps, cette partie a pour but de présenter les différentes catégories de modélisation de surfaces mises en place :

- les surfaces polynomiales
- les surfaces de subdivision
- les surfaces par ondelettes
- les maillages.

Nous nous attacherons à illustrer chaque catégorie par une ou plusieurs méthodes. Dans un second temps, nous dégagerons grâce à cette étude les avantages et inconvénients de chaque catégorie afin de déterminer les exigences auxquelles nous devons répondre dans la construction de notre modèle ainsi que les écueils à éviter.

Le lecteur est supposé familier des notions de base de CAGD (courbes de Bézier, courbes B-Splines, patches triangulaires de Bézier, carreaux de Bézier, B-Splines produits tensorielles). Dans le cas contraire, il peut se référer à [Farin, 1996] ou [Hoschek *et al.*, 1993].

## 1.1 Surfaces polynomiales

### 1.1.1 Présentation

Les surfaces polynomiales sont définies par une fonction polynomiale  $S$  d'un domaine planaire  $\Omega$  :

$$S : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}^3. \quad (1.1.1)$$

Une surface est rarement constituée d'un seul polynôme. On définit donc plusieurs fonctions polynomiales  $S_i$  sur plusieurs domaines  $\Omega_i$ . Ces derniers sont alors étendus

comme sous-ensembles plan de  $\mathbb{R}^3$  mais doivent constituer une variété de dimension deux dans  $\mathbb{R}^3$ . Les fonctions  $S_i$  définissent une surface par morceaux, chaque « patch » est l'image par une application de la facette plane  $\Omega_i$  qui l'a générée. Ces différents morceaux de surfaces doivent néanmoins se raccorder de manière continue et lisse pour constituer une seule surface. Pour qu'il n'y ait pas de trou, les images d'une arête par les deux fonctions polynomiales qui y sont définies doivent être les mêmes. Pour qu'il n'y ait pas d'angle, ces surfaces doivent se prolonger naturellement, i.e. avoir un plan tangent commun. C'est ce que l'on appelle la continuité géométrique d'ordre 1. Ces notions de continuité sont vues plus en détails dans les annexes (annexe A.1) et le lecteur intéressé pourra s'y reporter. Cette propriété de continuité géométrique a fait l'objet de nombreux travaux parmi lesquels [DeRose, 1990], [Peters, 2002b] et [Watkins, 1988]. Des continuités plus fortes peuvent être demandées mais ne sont pas nécessaires, voir même deviennent dépendantes de la paramétrisation.

Ces surfaces polynomiales sont classées suivant leur ensemble de définition, i.e. les  $\Omega_i$ . On distingue ainsi les surfaces de type produit tensoriel de celles définies sur un maillage triangulaire.

### 1.1.2 Produit Tensoriel

Les surfaces de type produit tensoriel sont définies sur une grille régulière du plan. Elles comprennent notamment les carreaux de Bézier, les B-splines. Elles sont obtenues à partir des courbes définies le long de la grille régulière. Soient  $(P_{i,j})_{0 \leq i \leq n_1, 0 \leq j \leq n_2}$  les points de contrôle de ces courbes. Elles sont définies par

$$C_k^1(u) = \sum_{i=0}^{n_1} P_{i,k} F_i(u) \quad (1.1.2)$$

$$C_k^2(v) = \sum_{j=0}^{n_2} P_{k,j} G_j(v) \quad (1.1.3)$$

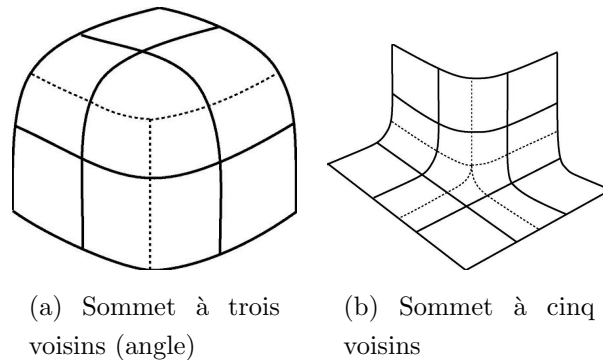
selon leur orientation (les  $F_i$  et  $G_i$  sont les fonctions de base).

**Définition 1.1.1** *La surface*

$$S(u, v) = \sum_{i=0}^{n_1} \sum_{j=0}^{n_2} P_{i,j} F_i(u) G_j(v) \quad (1.1.4)$$

où les paramètres  $u$  et  $v$  parcourent un domaine rectangulaire  $([0, 1]^2$  sans perte de généralité) est appelée surface produit tensoriel.

Ce type de surface est le plus utilisé en C.A.O., en particulier les B-splines cubiques produit tensoriel, qui sont des surfaces produit tensoriel avec des fonctions b-splines de degré trois comme fonctions de base. Ces surfaces présentent néanmoins quelques inconvénients. En effet, le principe d'utiliser une grille régulière s'avère très contraignant. Certaines configurations ne sont pas représentables par des grilles régulières (voir figure 1.1.1), et introduisent des sommets de valence (nombre de voisins dans le maillage) différente de quatre ou des faces avec un nombre d'arêtes différent de quatre.



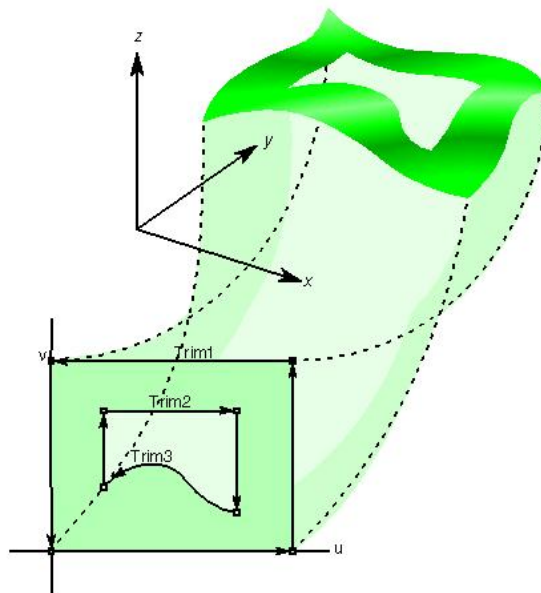
**Fig. 1.1.1** – Configurations non produit tensoriel

La surface doit alors être traitée différemment autour de ces sommets ou faces extraordinaires. [Loop et DeRose, 1989] introduit les S-patches pour résoudre ces difficultés. Les S-patches généralisent la notion de surface de Bézier produit tensoriel aux faces à n'importe quel nombre d'arêtes.

Par ailleurs, l'utilisateur contrôle la surface à travers les points de contrôle. Ces points de contrôle n'appartiennent généralement pas à la surface, ce qui rend leur utilisation peu intuitive. De plus les contraintes de raccordement impliquent des dépendances entre les points de contrôle de part et d'autre d'une frontière entre deux patches dans le cas de surfaces de Bézier, ce qui s'avère délicat à manipuler.

Pour permettre la modélisation de surfaces complexes, des courbes de découpe (trim-curves) sont aussi utilisées, ce qui complique l'affichage et la manipulation (voir figure 1.1.2).

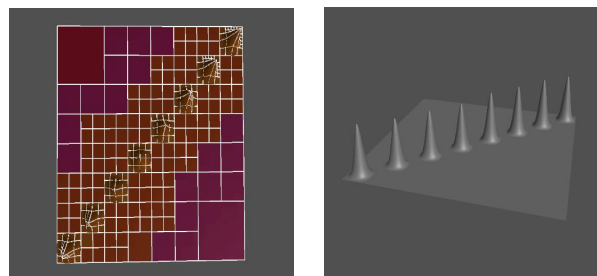
L'utilisation de ces surfaces de type produit tensoriel implique de représenter un objet à une précision donnée : Il n'est pas possible d'éditer plus finement un unique carreau sans que l'ensemble de l'objet ne se retrouve raffiné. Cela complique et alourdit inutilement la représentation de l'objet. [Sederberg *et al.*, 2003] introduit les T-Splines qui permettent de gérer une connexion en forme de « T » au sein de l'espace des paramètres. [Forsey et Bartels, 1988] résolvent quant à eux ce problème par la notion de hiérarchie avec les H-splines.



**Fig. 1.1.2** – Modélisation par courbes de découpe (trim curves) - Images extraites du livre « OpenGL Optimizer Programmer's Guide : An Open API for Large-Model Visualization »

### H-Splines

Les H-splines répondent au besoin de pouvoir ajouter localement des détails sans intervenir sur le reste de la surface. Pour ce faire, [Forsey et Bartels, 1988] apporte



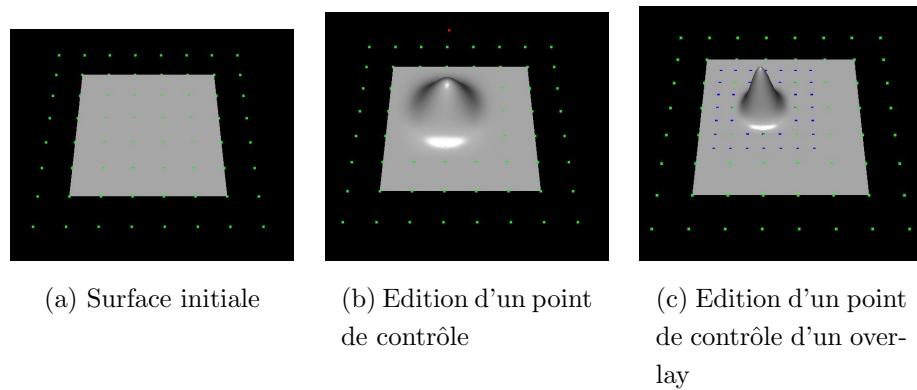
(a) Subdivision locales

(b) Edition le long de la diagonale

**Fig. 1.1.3** – Raffinement le long de la diagonale : cas le pire. Sans les overlays, tout devrait être subdivisé - Images extraites de la page web de David Forsey

les *overlays*, qui sont des carreaux de splines plus fins définis localement relativement au carreau de spline sous-jacent. Cela permet de préciser la surface en ajoutant des détails à la taille voulue sans raffiner le reste de la surface (voir figures 1.1.3 et 1.1.4). De plus, un principe de référencement par rapport à la surface moins raffinée

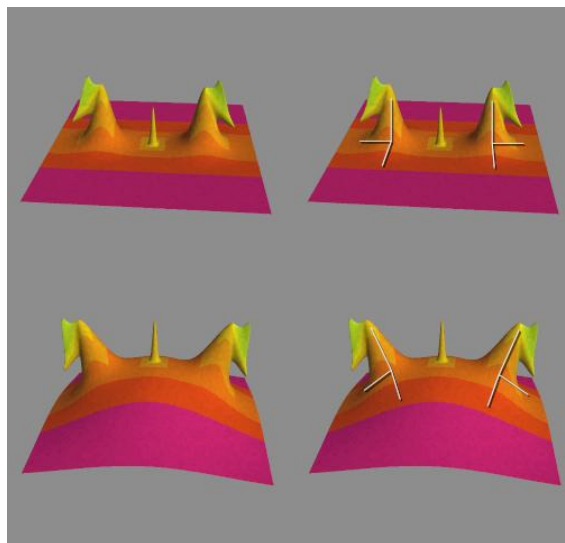




**Fig. 1.1.4** – Définition d'un overlay : édition avant et après ajout d'un overlay - Images extraites du site web de Wayne Olive

sous jacente (surface mère) facilite l'édition. En effet, chaque point de contrôle  $q_{i,j}$  de l'overlay est défini dans un repère local  $\mathcal{R}_{i,j}$  dépendant de la surface mère. Il est centré au point  $r_{i,j}$  de la surface le plus influencé par  $q_{i,j}$  et orienté selon la normale et le plan tangent à la surface en  $r_{i,j}$ . Le point  $q_{i,j}$  s'écrit donc  $q_{i,j} = r_{i,j} + o_{i,j}$  avec  $o_{i,j}$  vecteur de déplacement (offset) exprimé dans  $\mathcal{R}_{i,j}$ .

Un détail est donc inséré en déplaçant  $q_{i,j}$  par une modification de  $o_{i,j}$ . A l'inverse, lors d'une édition globale, la surface mère est modifiée et donc  $\mathcal{R}_{i,j}$  aussi.  $q_{i,j}$  est donc mis à jour directement par  $r_{i,j}$ , tout en conservant les détails à travers  $o_{i,j}$ . Les détails suivent donc naturellement la déformation globale (voir figure 1.1.5).



**Fig. 1.1.5** – Déformation globale avec conservation des détails - Images extraites du site web de David Forsey

Un autre apport de [Forsey et Bartels, 1988] est l'introduction de points d'édition. Ce sont des points de la surface pouvant être déplacés. Ils correspondent à un maximum d'influence d'un point de contrôle et transmettent leur déplacement directement à celui-ci.

Les H-Splines créent une hiérarchie de détails permettant une approche plus intuitive pour représenter des surfaces : une surface simple qui sert de brouillon et à laquelle on ajoute des détails pour obtenir l'objet complet. De plus, l'édition globale facilite l'animation en gérant les déformations globales tout en conservant les détails. Par ailleurs, cette hiérarchie est utilisée pour le rendu à travers les techniques d'affichage multirésolution. [Forsey et Bartels, 1995] et [Forsey et Wong, 1998] mettent en œuvre les H-splines pour reconstruire des objets grâce à des techniques de minimisation. Les H-splines sont donc très attrayantes car elles ajoutent à des outils très utilisés une notion de multirésolution, d'édition globale ainsi que d'édition directe de la surface. Le principal problème reste toujours d'actualité : seules des topologies de type produit tensoriel peuvent être manipulées.

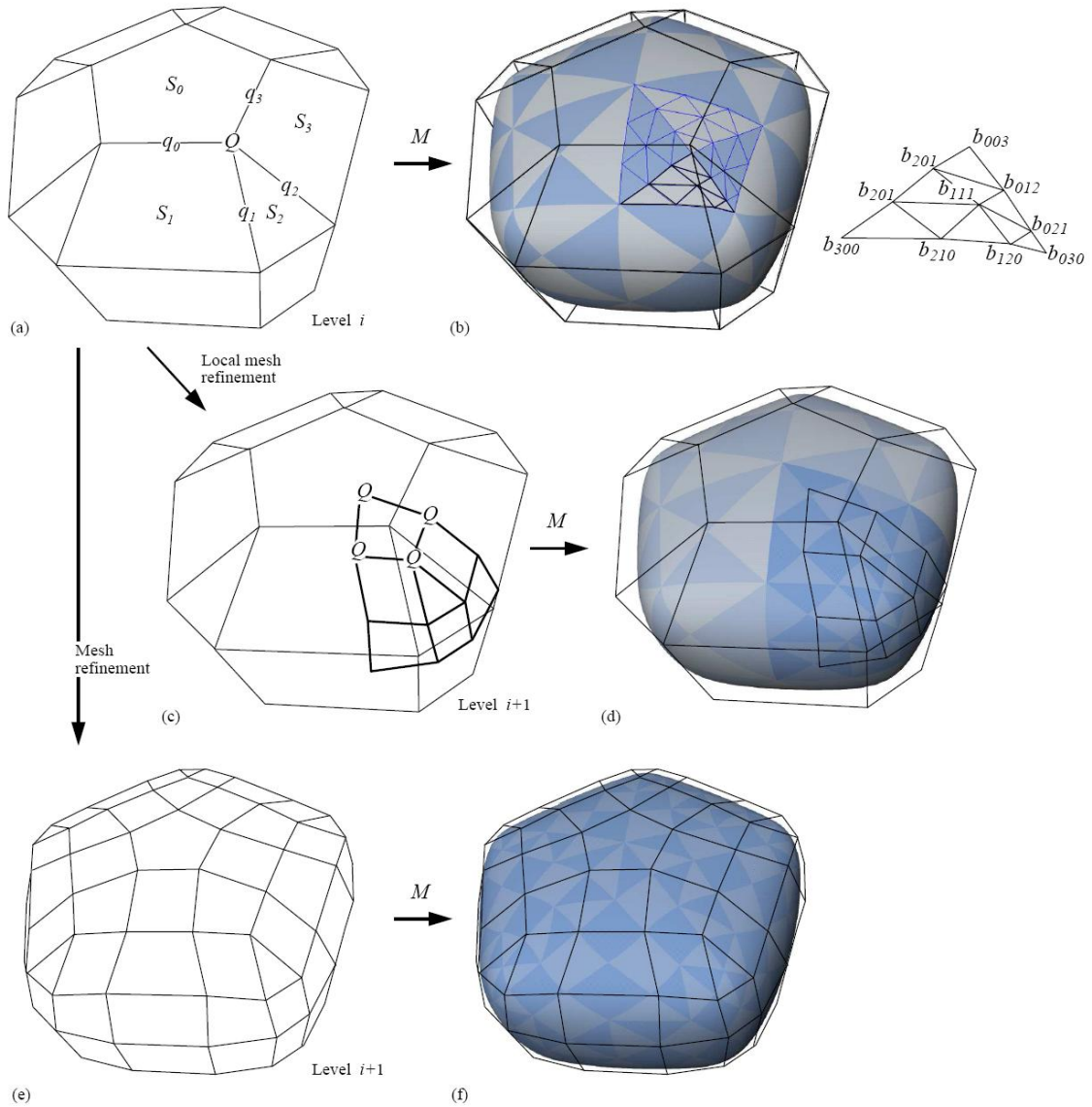
### Localized-Hierarchy Surface Splines (LeSS)

[Gonzalez-Ochoa et Peters, 1999] remarque que [Forsey et Bartels, 1988] ne traite que les configurations produit tensoriel et que les calculs peuvent devenir instables avec la multiplication des niveaux. Les Localized-Hierarchy Surface Splines (LeSS) sont donc développées. Elles sont définies sur un maillage dont tous les sommets ont exactement quatre voisins. Un pré-calcul est proposé pour transformer n'importe quel maillage en un maillage respectant cette contrainte. Une hiérarchie de maillages (par opposition à hiérarchie de surfaces) est créée. Une surface est directement calculée sur le maillage le plus fin. Il est important de remarquer que la surface n'est pas elle-même hiérarchique. Cette surface est construite sous forme de triangles de Bézier. La figure 1.1.6 montre un calcul de surface ainsi qu'un raffinement local et un global.

L'intérêt de cette hiérarchie de maillages plutôt que de surfaces est la possibilité de changer la topologie en cours d'édition (voir figure 1.1.7). Par ailleurs, l'édition se fait en manipulant directement un point de la surface ; les points de contrôle sont alors modifiés en correspondance.

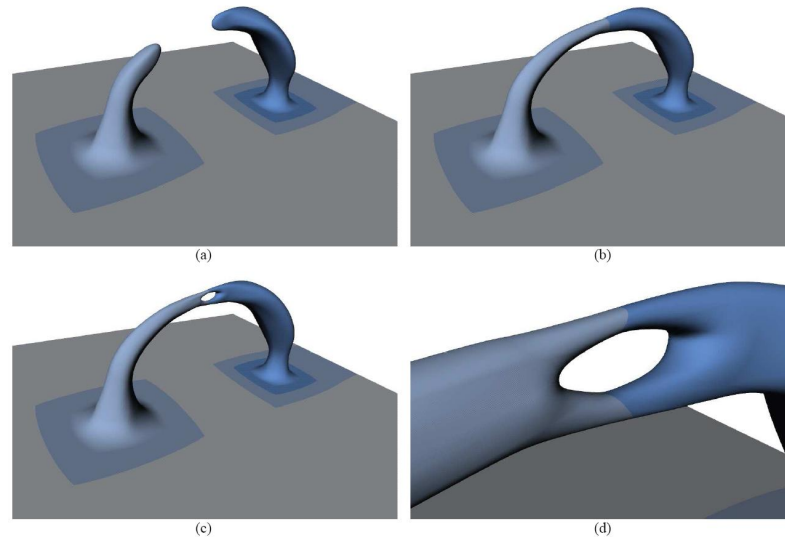
### 1.1.3 Maillages triangulaires

Pour pouvoir manipuler toutes les topologies, la solution la plus simple consiste à utiliser des triangulations.



**Fig. 1.1.6** – (a) Exemple de maillage. Les 9 points indiqués définissent quatre patches triangulaires de Bézier (b). (c) Un raffinement local et sa surface (d). (e) Raffinement global et sa surface (f) - Images extraites de [Gonzalez-Ochoa et Peters, 1999].

On note  $V$  un ensemble de points de  $\mathbb{R}^3$ . Un simplexe de dimension 0, 1, ou 2 est l'ensemble des combinaisons convexes respectivement de 1, 2 ou 3 éléments de  $V$ , qui sont appelés les sommets de ce simplexe. On dit qu'un simplexe  $\sigma$  est face d'un simplexe  $\tau$  si et seulement si les sommets de  $\sigma$  sont aussi sommets de  $\tau$ . Une triangulation  $K$  est un ensemble de simplexes de dimension 0, 1 ou 2 avec les propriétés suivantes :



**Fig. 1.1.7** – Exemple de changement de topologie. Deux protubérances (a) fusionnent en une anse (b). Un trou est alors percé (c) (d) - Images extraites de [Gonzalez-Ochoa et Peters, 1999].

(i)  $\sigma \in K$  et  $\tau$  est une face de  $\sigma \Rightarrow \tau \in K$

(ii)  $\sigma, \tau \in K \Rightarrow \sigma \cap \tau$  est une face de  $\sigma$  et de  $\tau$

Une variété à bord de dimension 2 est un espace topologique dans lequel le voisinage de tout point est homéomorphe soit à  $\mathbb{R}^2$ , soit à  $\mathbb{R}^+ \times \mathbb{R}$ .

Par extension, on dit qu'une triangulation  $K$  est une variété à bord de dimension 2 si et seulement si l'union de l'ensemble des simplexes de  $K$ , muni de la topologie induite de celle de  $\mathbb{R}^3$ , est une variété à bord de dimension 2.

Dans cette thèse, on ne traite que de triangulations qui sont des variétés à bord de dimension 2. On les nomme sans ambiguïté triangulations surfaciques, ou simplement triangulations. Par ailleurs les faces à deux sommets sont appelées arêtes. Le terme face est ainsi réservé aux simplexes à trois sommets, i.e. aux triangles.

Sur ces triangulations, des surfaces polynomiales sont construites. A chaque face de la triangulation est associé un patch polynomial, i.e. un morceau de surface lisse. Il existe beaucoup de méthodes de calcul de surfaces lisses sur une triangulation. Nous nous concentrons particulièrement sur les schémas interpolants, i.e. qui passent par les sommets de la triangulation. Cela permet un meilleur contrôle de la surface puisque l'utilisateur déplace ainsi directement des points de la surface. [Mann *et al.*, 1992] et [Peters, 1990b] font un bref état de l'art de ces divers interpolants, mais qui sont déjà anciens. Par ailleurs, seules les méthodes générant des surfaces lisses sont envisagées. Par lisse, nous entendons de continuité géométrique

d'ordre 1, i.e. continuité du plan tangent. Dans une première partie, les implications de cette continuité géométrique sont rappelées, en particulier le problème de consistance aux sommets du maillage. Après quoi plusieurs stratégies de construction de surfaces lisses interpolant un maillage triangulaire sont vues, triées par leur solution au problème de consistance au sommet :

- Stratégie par subdivision de domaine
- Stratégie par combinaison convexe
- Stratégie par facettes dégénérées
- Stratégie par courbes frontières.

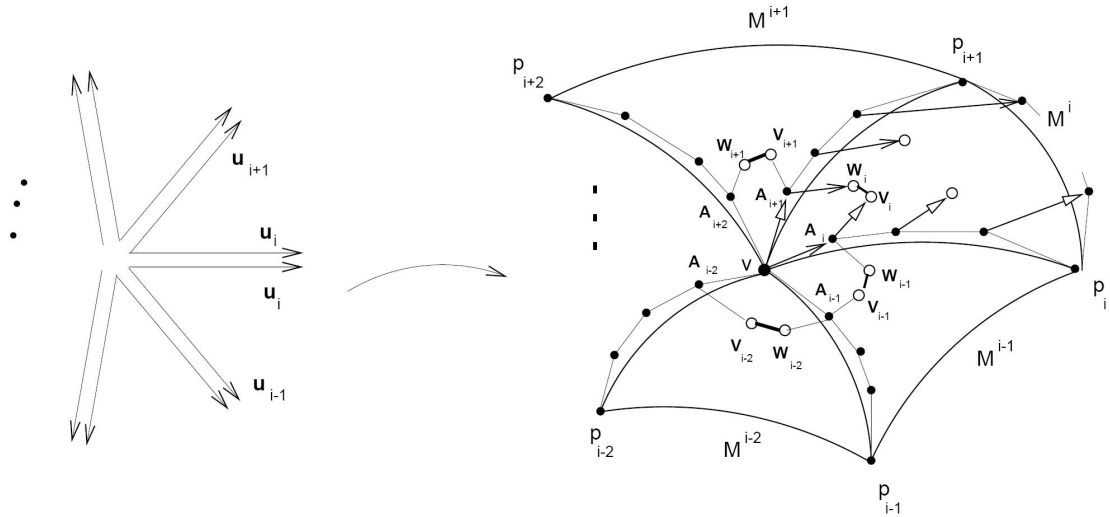
Seules des méthodes locales sont présentées. La propriété de localité signifie que la construction d'un patch de la surface ne nécessite de connaître que ses voisins. Cette propriété s'avère extrêmement utile lors d'une modification de la surface : seul le voisinage de la partie modifiée est recalculé, ce qui est important lors de la manipulation de maillages de plusieurs centaines de millier de faces.

Il convient aussi de mentionner les méthodes dites de « scattered data ». Ces méthodes traitent des données de la forme  $(x_i, y_i, f_i)$ ,  $x_i$ ,  $y_i$  et  $f_i$  réels, pour  $i = 0, \dots, n$ . Leur but est de créer une fonction définie sur un domaine  $D$  de  $\mathbb{R}^2$  ( $f : D \subset \mathbb{R}^2 \rightarrow \mathbb{R}$ ) telle que  $f(x_i, y_i) = f_i$  pour tout  $i$ . Une triangulation planaire sous-jacente peut aussi être donnée, comme par exemple dans le cas du *minimum norm network* [Nielson, 1980, Nielson, 1983]. Ces méthodes existent aussi en 3 dimensions. [Alfeld, 1989, Franke, 1982, Franke et Nielson, 1990] constituent des synthèses. Nous ne développerons pas plus cet aspect car il est différent du sujet de cet ouvrage.

### Problème de la consistance aux sommets

En un sommet où plusieurs courbes frontières des patches incidents à ce sommet se rejoignent, celles-ci ne peuvent pas être quelconque si l'on veut pouvoir construire une surface lisse les interpolant. Plus précisément, leurs dérivées en ce sommet ne peuvent pas être quelconques. C'est le problème de la consistance en un sommet. Il est nécessaire d'obtenir le raccordement avec continuité géométrique d'ordre un ( $G^1$ ) le long d'une courbe de bord (cf. annexe A.1). Dans le cas d'une surface définie par patches triangulaires de Bézier, la surface est spécifiée par des points dans l'espace. Ces points représentent les coefficients du polynôme dans la base de Bézier. Cette représentation présente de nombreux avantages, entre autres de donner un aperçu de la surface en reliant ces points de contrôle. La continuité géométrique d'ordre un impose des contraintes sur les points de contrôle de la courbe frontière entre deux patches ainsi que ceux situés sur la première rangée intérieure. Un objet est

composé de beaucoup plus que de deux patches. Il faut alors considérer ce qu'il se passe en un sommet  $v$  où  $n$  patches de Bézier  $M_i$ ,  $i = 1 \dots n$  se raccordent. Si



**Fig. 1.1.8** – Problème de consistance en un sommet

on se contente de raccorder les facettes consécutives deux à deux, le plan tangent le long de chaque courbe frontière est construit, et donc les points de contrôle des deux premières rangées de chaque patch. Or le point de contrôle situé dans l'angle appartient à deux rangées intérieures : celle correspondant à chaque courbe incidente au sommet. Avec les notations de la figure 1.1.8, le point de contrôle intérieur  $V_i$  situé dans l'angle du patch  $M_i$  sert à la définition du plan tangent le long de deux courbes frontières. Pour raccorder les patches de manière  $G^1$  on calcule donc  $V_i$  lors du raccord de  $M_{i+1}$  et  $M_i$ . Un second point de contrôle  $W_i$  est calculé lors du raccord de  $M_{i-1}$  et  $M_i$ . Or ces points sont censés être identiques puisqu'il n'existe qu'un point de contrôle dans l'angle. Mais rien n'impose à  $W_i$  d'être identique à  $V_i$  dans cette démarche itérative. Mais il doit l'être puisque le patch est polynomial, donc de classe  $C^2$ . Par conséquent, ses dérivées secondes mixtes doivent être égales au sommet.

Dans les faits, il n'est tout simplement pas possible d'obtenir deux points identiques si les dérivées ont été placées sans contrainte. En effet, si l'on impose que  $V_i = W_i$  pour tout  $i$ , alors pour  $V_0$  fixé,  $V_1$  est obtenu grâce à la contrainte de continuité  $G^1$ .  $V_1$  fixé,  $V_2$  est obtenu grâce à la contrainte de continuité  $G^1$ , et ainsi de suite. On arrive ainsi à boucler autour du sommet et à en déduire  $V_0$ , qui était fixé. Le problème est donc de trouver un point  $V_0$  qui soit invariant après calculs autour du sommet.

D'un point de vue mathématique, le vecteur  $T$  des points  $V_i$  autour du sommet

est lié au vecteur  $D$  des points des dérivées autour de ce sommet par une équation de la forme :

$$M.T = N.D \quad (1.1.5)$$

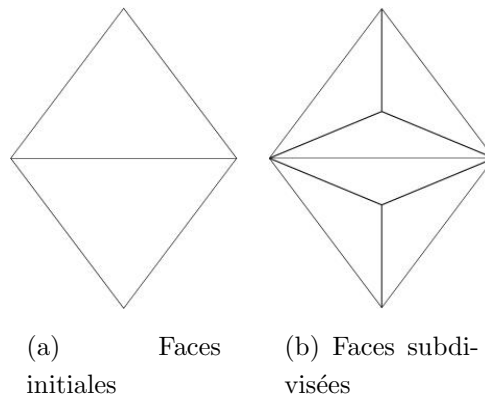
où  $M$  et  $N$  sont des matrices.  $T$  étant l'inconnu, ce système possède une solution quel que soit  $D$  si et seulement si  $M$  est inversible, ce qui n'est pas le cas généralement. Différentes stratégies sont donc appliquées pour obtenir des points  $V_i$  et  $A_i$  vérifiant ce système.

Les points  $V_i$  situés dans l'angle des patches sont appelés points twists des facettes.

Le problème d'obtenir des points twists vérifiant l'équation (1.1.5) s'appelle le problème de consistance au sommet. Plusieurs approches ont été proposées pour résoudre ce problème, parmi lesquelles les stratégies par subdivision du domaine, par combinaison convexe, par facettes dégénérées, par facettes algébriques ou par courbes frontières.

### Stratégie par subdivision du domaine

L'une des solutions pour résoudre le problème de consistance aux sommets est de séparer les contraintes autour d'un sommet en construisant plusieurs patches de surface par face. Pour cela, une subdivision de type Clough-Tocher (voir figure 1.1.9) du domaine introduit suffisamment de degré de liberté pour satisfaire les contraintes autour du sommet. Cette technique est illustrée ci-dessous par les méthodes de

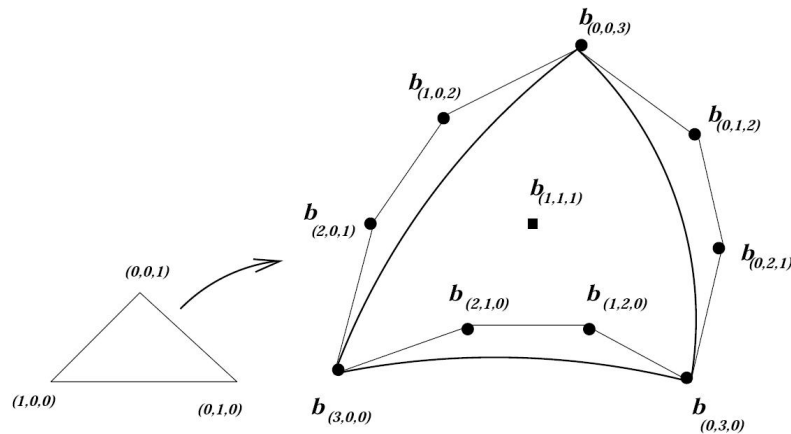


**Fig. 1.1.9** – Subdivision de Clough-Tocher

[Piper, 1987], [Shirman et Sequin, 1987] et [Jensen, 1987].

**Méthode de Piper** [Piper, 1987] propose un interpolant quartique basé sur une subdivision de type Clough-Tocher. Un seul patch cubique interpolant est d'abord

calculé par face. Il est alors subdivisé au centroïde\*, puis subit une élévation au degré quatre. Les points de contrôle de ces nouveaux patches sont alors ajustés pour obtenir une continuité  $G^1$  le long des courbes frontières extérieures, et une continuité  $C^1$  le long des courbes frontières intérieures. Les points de contrôle de la face cubique sont



**Fig. 1.1.10** – Face de Bézier cubique

déterminés ainsi :  $b_{(3,0,0)}$ ,  $b_{(0,3,0)}$  et  $b_{(0,0,3)}$  sont les sommets de la face car le schéma est interpolant. Le point  $b_{(2,1,0)}$  est obtenu en projetant  $b_{(0,3,0)}$  dans le plan tangent en  $b_{(3,0,0)}$  (donné par son vecteur normal) et en multipliant le vecteur du point  $b_{(3,0,0)}$  au point projeté par un constante ( $\frac{4}{9}$  de la norme du vecteur). Les autres points le long des courbes sont obtenus de manière symétrique. Le point restant ( $b_{(1,1,1)}$ ) est calculé comme une combinaison linéaire de ces points. Cette technique assure que chaque patch sera lisse, mais pas la jonction entre les patches. Pour cela, chaque patch est subdivisé en son centroïde. Les points de contrôle intérieurs sont placés pour minimiser les ondulations. Ceux le long des courbes intérieures sont modifiés pour raccorder les sous patches de manière lisse. Chaque patch est alors élevé au degré quatre. Les nouveaux points de contrôle le long des courbes frontières sont finalement calculés pour raccorder les patches de manière  $G^1$ . Comme cette solution n'est pas unique, [Piper, 1987] choisit la solution la plus proche des points avant modification. Trois patches de degré quatre se raccordant de manière  $C^1$  sont alors construits par face, la continuité entre deux patches appartenant à deux faces différentes est  $G^1$ .

**Méthode de Shirman Sequin** [Shirman et Sequin, 1987] et [Shirman et Sequin, 1991] appliquent la méthode de Chiyokura-Kimura (voir annexe A) à trois patches triangulaires de degré quatre résultant d'une subdivision de type Clough-Tocher. Cette

\*le centroïde est l'image du point de paramètre  $(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$



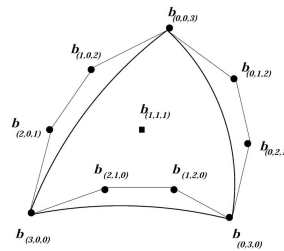
construction suppose que des courbes frontières cubiques sont données et qu'elles ont été élevées au degré quatre. Les points intérieurs restants sont alors directement calculés par la méthode de Chiyokura-Kimura pour la continuité  $G^1$  le long des frontières extérieures et par la méthode de Farin le long des courbes de bord intérieures.

**Méthode de Jensen** [Jensen, 1987] ressemble beaucoup à [Shirman et Sequin, 1987] à la différence qu'une continuité  $C^1$  est obtenue le long des courbes de bord intérieures, et non plus  $G^1$ . Par ailleurs, une version modifiée de Chiyokura-Kimura est utilisée pour raccorder deux facettes.

### Stratégie par Combinaison convexe

Les schémas par combinaison convexe créent un seul patch par face. Ces patches sont de continuité  $C^2$  partout sauf aux sommets. Grâce à cela, le problème de consistance aux sommets est résolu puisque le twist n'est pas continu. Ces méthodes construisent en premier des courbes de bord ainsi que les plans tangents le long de celles-ci. Trois patches sont alors générés qui interpolent une partie de ces données. Finalement, un seul patch interpolant toutes ces données est produit par combinaison convexe des trois précédents. Ces méthodes sont illustrées par les patches de Gregory triangulaires.

**Patches triangulaires de Gregory** L'idée des patches de Gregory est de considérer chaque twist comme un mélange de deux twists, un pour chaque courbe de bord incidente au sommet considéré. Un réseau de courbes de bord cubiques ainsi que leur plan tangent est d'abord construit par la méthode de Chiyokura-Kimura (voir annexe A). Six points de contrôle intérieurs  $P_{(i,j)}$  sont alors obtenus (voir figure 1.1.11). Néanmoins, ces points ne vérifient pas la contrainte des twists. Pour résoudre



**Fig. 1.1.11** – Patch de Gregory

ce problème, une facette quartique rationnelle à twists discontinus est construite :

$$M(\lambda_1, \lambda_2, \lambda_3) = \sum_{p+q+r=4} B_{(p,q,r)}^4(\lambda_1, \lambda_2, \lambda_3) b_{(p,q,r)} \quad (1.1.6)$$

où  $(\lambda_1, \lambda_2, \lambda_3)$  sont les coordonnées barycentriques d'un point, et  $b_{(p,q,r)}$  les points de contrôle de la surface. Les points twists  $b_{(1,1,2)}$ ,  $b_{(1,2,1)}$  et  $b_{(2,1,1)}$  sont obtenus par une combinaison convexe des points obtenus par Chiyokura-Kimura :

$$\begin{cases} b_{(1,1,2)} = \frac{\lambda_1(1-\lambda_2)p_{(3,1)} + \lambda_2(1-\lambda_1)p_{(3,2)}}{\lambda_1(1-\lambda_2) + \lambda_2(1-\lambda_1)} \\ b_{(1,2,1)} = \frac{\lambda_1(1-\lambda_3)p_{(2,1)} + \lambda_3(1-\lambda_1)p_{(2,3)}}{\lambda_1(1-\lambda_3) + \lambda_3(1-\lambda_1)} \\ b_{(2,1,1)} = \frac{\lambda_3(1-\lambda_2)p_{(1,3)} + \lambda_2(1-\lambda_3)p_{(1,2)}}{\lambda_3(1-\lambda_2) + \lambda_2(1-\lambda_3)} \end{cases} \quad (1.1.7)$$

Cette facette est donc obtenue simplement à partir de la méthode de Chiyokura-Kimura. Néanmoins il faut remarquer qu'elle peut être interprétée comme une combinaison convexe de sept patches de Bézier de degré quatre avec des facteurs sous forme de fonctions rationnelles de polynômes de degré six. En pratique donc, ces degrés sont trop élevés pour être utilisés. Cette méthode résout les problèmes de consistance au sommet élégamment dans un cadre théorique, mais n'est pas exploitable dans la pratique.

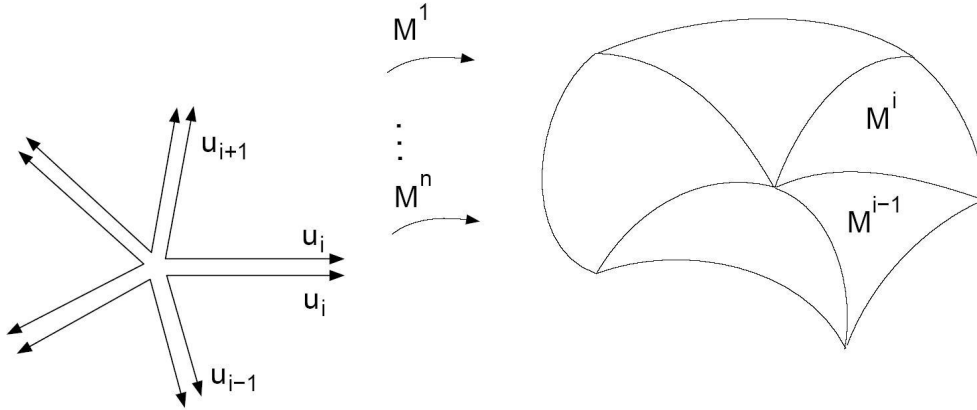
### Stratégie par facettes dégénérées

Une approche possible pour résoudre le problème de consistance au sommet est l'utilisation de facettes dégénérées. Celles-ci sont des facettes de Bézier dont les deux derniers points de contrôle des courbes de bord sont confondus : on dit qu'elles sont dégénérées d'un degré aux sommets. Cette propriété est suffisante pour assurer la consistance aux sommets. On peut alors construire une seule facette quintique dégénérée par face à l'aide d'une minimisation par exemple. Néanmoins leur utilisation pose des problèmes numériques.

### Stratégie par Courbes frontières

Les méthodes vues précédemment calculent d'abord des courbes frontières ; les patches sont ensuite remplis pour se raccorder de manière  $G^1$  et résoudre le problème de consistance. Il est toutefois possible de prendre en compte cette contrainte dans le calcul de ces courbes et de générer des courbes de bord  $C^2$  compatibles. Les méthodes de Loop [Loop, 1994], Peters [Peters, 1991] et Hahmann Bonneau [Bonneau et Hahmann, 2000] utilisent cette approche.

**Méthode de Loop** [Loop, 1994] fait correspondre un patch de Bézier de degré six par face. Cette méthode a été conçue pour approcher un maillage, mais l'interpolation est également possible. Néanmoins, celle-ci mène à des ondulations indésirées ce qui en limite l'application.



**Fig. 1.1.12** – Paramétrisation autour d'un sommet.

Autour d'un sommet  $v$ ,  $n$  patches  $M_i$  sont créés pour  $i = 1, \dots, n$ . Avec les notations de la figure 1.1.12, rappelons que deux patches  $M_{i-1}(u_{i-1}, u_i)$  et  $M_i(u_i, u_{i+1})$  se raccordent de manière  $G^1$  si

$$M_{i-1}(0, u_i) = M_i(u_i, 0) \quad \forall u_i \in [0, 1] \quad (1.1.8)$$

et il existe trois fonctions réelles  $\Phi_i$ ,  $\mu_i$  et  $\nu_i$  telles que

$$\Phi_i(u_i) \frac{\partial M_i}{\partial u_i}(u_i, 0) = \mu_i(u_i) \frac{\partial M_{i-1}}{\partial u_{i-1}}(0, u_i) + \nu_i(u_i) \frac{\partial M_i}{\partial u_{i+1}}(u_i, 0) \quad (1.1.9)$$

$$\frac{\partial M_i}{\partial u_i}(u_i, 0) \cdot \frac{\partial M_i}{\partial u_i}(u_i, 0) \neq 0 \quad (1.1.10)$$

$$\mu_i(u_i) \cdot \nu_i(u_i) > 0 \quad \forall u_i \in [0, 1]. \quad (1.1.11)$$

Si l'on dérive l'équation (1.1.9) en fonction de  $u_i$  puis on l'évalue en  $u_i = 0$ , on

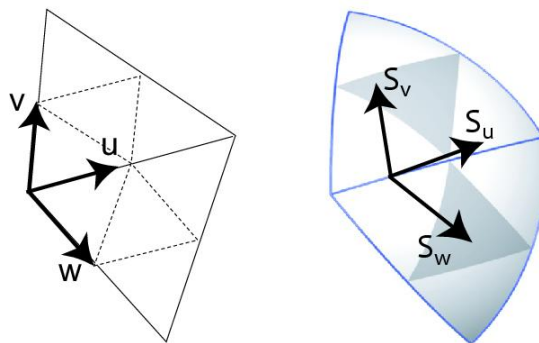
obtient le système

$$\begin{aligned} \mu_i(0) \frac{\partial^2 M_{i-1}}{\partial u_{i-1} \partial u_i}(0,0) + \nu_i(0) \frac{\partial^2 M_i}{\partial u_i \partial u_{i+1}}(0,0) = \\ \Phi'_i(0) \frac{\partial M_i}{\partial u_i}(0,0) + \Phi_i(0) \frac{\partial^2 M_i}{\partial u_i^2}(0,0) + \\ \nu'_i(0) \frac{\partial M_i}{\partial u_{i+1}}(0,0) + \mu'_i(0) \frac{\partial M_{i-1}}{\partial u_{i-1}}(0,0) \end{aligned} \quad (1.1.12)$$

$$i = 1, \dots, n.$$

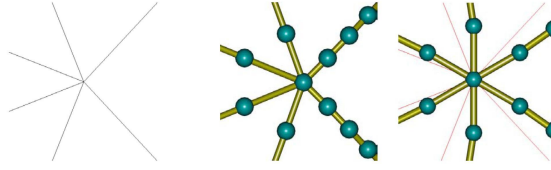
Le second membre est fixé pour des courbes frontières données. Ainsi, si les courbes frontières ont été calculées lors d'une première étape, on peut en déduire les twists en inversant le système 1.1.12. Le problème est que ce système possède une matrice circulante qui n'est pas inversible si  $n$  est pair. Loop propose donc de construire des courbes frontières appartenant à l'espace image de cette matrice. Il existe alors toujours une solution au système.

**Méthode de Hahmann-Bonneau** [Bonneau et Hahmann, 2000] est basé sur la méthode de Loop vue ci-dessus. L'apport principal est de la rendre interpolante et de baisser son degré à cinq. Pour cela, [Bonneau et Hahmann, 2000] fait correspondre quatre patches de Bézier à chaque face de la triangulation selon la règle de la subdivision en quatre. Chaque face est découpée en quatre faces filles obtenues en insérant des sommets au milieu des arêtes et en les reliant (voir figure 1.1.13).



**Fig. 1.1.13** – Découpage de l'espace des paramètres en quatre.

L'introduction de cette subdivision crée de nouveaux degrés de liberté qui sont utilisés pour baisser le degré polynomial de la surface. De plus, les différentes fonctions sont alors définies par morceaux ce qui rend possible l'interpolation.



**Fig. 1.1.14** – Courbes de bord autour d'un sommet. Au milieu, courbes issues de [Hahmann et Bonneau, 2003]. Sur la droite, les courbes sont issues de [Bonneau et Hahmann, 2000].

[Hahmann et Bonneau, 2003] apporte des améliorations à cette méthode. En effet, la résolution de la consistance aux sommets forçait les vecteurs dérivées premières à être l'image du polygone régulier de même ordre (voir figure 1.1.14). Cette contrainte était gênante dans le cas de maillages présentant de grandes faces voisines de petites. Cette contrainte a donc été enlevée dans [Hahmann et Bonneau, 2003] en prenant les twists comme paramètres libres de la construction.

**Méthode de Peters** [Peters, 1991] propose de résoudre le problème de consistance au sommet en ajoutant une contrainte sur les courbes : qu'elles soient compatibles avec une 2<sup>ème</sup> forme fondamentale donnée en chaque sommet par deux courbures principales et une direction principale. A l'aide de cette condition, il trouve une expression des fonctions réelles  $\Phi_i$ ,  $\mu_i$  et  $\nu_i$  assurant aux dérivées secondes des courbes de rester dans l'espace image généré par les twists dans l'équation (1.1.12). Les courbes de bord sont construites de degré trois comme avec Piper dans un premier temps. Une élévation de degré est ensuite effectuée, puis les points de contrôle sont modifiés pour prendre en compte la contrainte de courbure principale. Les fonctions réelles  $\Phi_i$ ,  $\mu_i$  et  $\nu_i$  sont alors calculées suivies du plan tangent le long des courbes de bord. Il est à noter qu'une subdivision de type Clough-Tocher est possible pour abaisser le degré.

#### 1.1.4 Conclusion

La modélisation par surface polynomiale demeure la plus utilisée dans le domaine de la CFAO. En effet, elle repose sur une expression analytique de la surface ce qui permet des calculs exacts. Deux grandes approches de surfaces polynomiales ont été étudiées : celle à base de produit tensoriel et celle à base de maillages triangulaires. Toutes deux présentent des avantages et des inconvénients notables.

Dans le cadre du produit tensoriel, ses points forts sont : son caractère répandu,

les possibilités d'interpolation et son implémentation aisée.

Néanmoins, cette approche ne s'adapte pas à toutes les topologies, et l'utilisation d'une grille régulière rend difficile l'ajout de détails hors des lignes de cette grille et nécessite l'usage de courbes de découpes.

De même, si le maillage triangulaire s'applique à toutes les topologies, il pose le problème de la consistance au sommet, souvent difficile à résoudre, ce qui implique un degré polynomial plus élevé qui dégrade les performances.

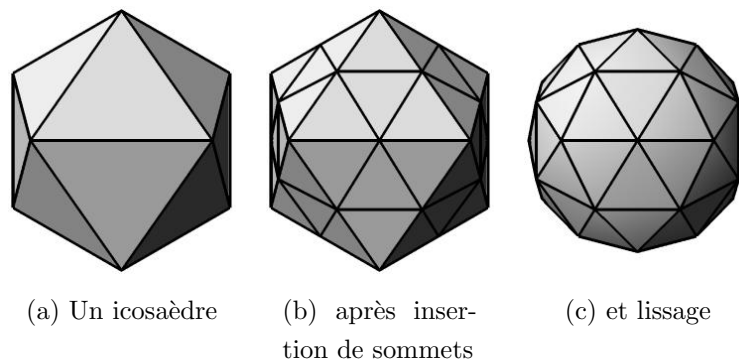
De plus, les surfaces rationnelles n'ont pas été abordées dans cette partie du fait de leur plus grande complexité de calcul ainsi que de l'attention nécessaire pour ne pas tomber dans des instabilités numériques.

En infographie, d'autres méthodes plus rapides sont préférées aux surfaces polynomiales car elles demeurent complexes à adapter dans un objectif de performance.

## 1.2 Surfaces de subdivision

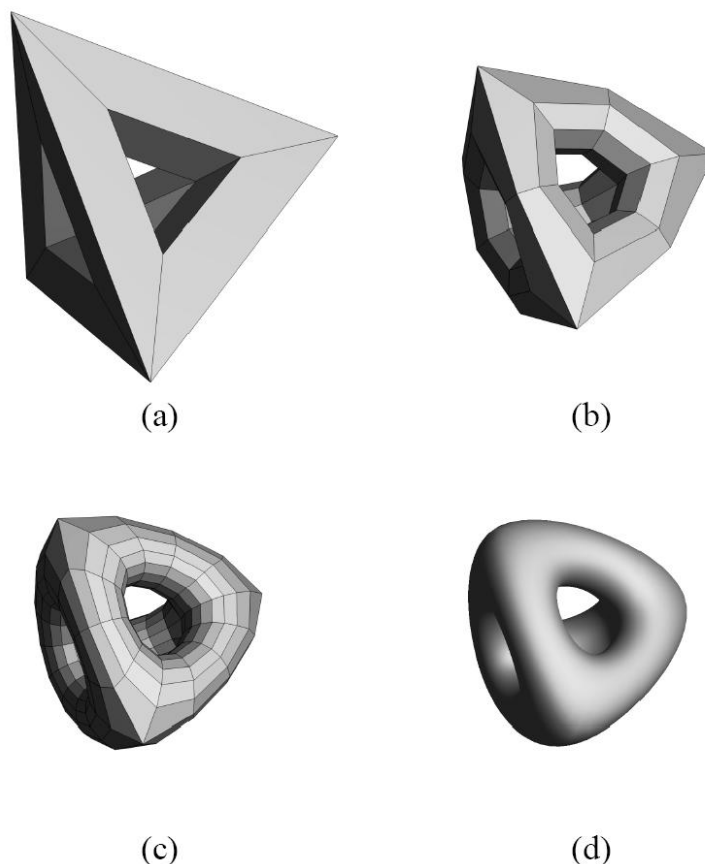
### 1.2.1 Présentation

Une surface de subdivision est définie comme la limite d'une séquence de raffinements successifs de maillages polyédriques. Un raffinement est composé de deux étapes : une opération topologique d'insertion de sommet (figure 1.2.1(b)) suivie d'une opération géométrique de lissage des sommets (figure 1.2.1(c)). On obtient



**Fig. 1.2.1** – Surfaces de Subdivision : une étape de raffinement

ainsi une suite de maillages polyédriques de plus en plus fins qui convergent vers une surface lisse : la surface de subdivision (voir figure 1.2.2 ). Ces surfaces ont été introduites dans les années 70 mais ont eu du mal à proliférer jusqu'à leur utilisation récente dans l'infographie. De nombreux schémas de subdivisions sont apparus ( Loop, Catmull-Clark, Doo-Sabin, butterfly. . . ). [Schröder et Zorin, 1998] constitue un état de l'art de la théorie et des principaux schémas de subdivision. [Schweitzer, 1996]

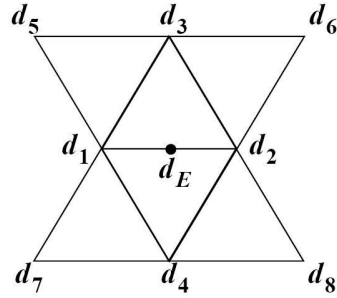


**Fig. 1.2.2** – Surfaces de Subdivision : (a) maillage de contrôle, (b) après une étape de subdivision, (c) après deux étapes de subdivision, (d) la surface de subdivision - Image extraite de [DeRose *et al.*, 1998]

regroupe une analyse théorique approfondie ainsi qu'une mise en pratique des surfaces de subdivision. Les surfaces de subdivision, bien qu'élégantes présentent parfois des résultats inattendus voire inexploitable. [Sabin et Barthe, 2003] présente des situations très régulières pour lesquelles les surfaces de subdivision résultantes ne correspondent pas aux attentes. En particulier, des maillages convexes réguliers génèrent des surfaces non convexes voire qui ondulent.

Parmi ces schémas, le schéma Butterfly ([Dyn *et al.*, 1990]) est un schéma interpolant, propriété qui s'avère intéressante. Pour qu'un schéma de subdivision soit interpolant, il faut que les sommets des maillages successifs soient sur la surface limite. Pour cela, il suffit de ne jamais déplacer un sommet après sa création. La surface générée est la limite des maillages successifs, donc si un sommet n'est jamais modifié par le schéma, il appartiendra à tous les maillages, et donc à la surface. Bien

que cette condition assure l'interpolation, elle ne suffit pas à garantir l'aspect lisse des surfaces engendrées. Bien évidemment, les sommets n'étant jamais modifiés, ils doivent immédiatement être correctement placés. Un mauvais placement lors des premières étapes de subdivision conduira à une surface bosselée. [Dyn *et al.*, 1990]



**Fig. 1.2.3** – Masque du schéma butterfly - Image extraite de [Hubeli et Gross, 2000]

produit des surfaces  $C^1$  dans les cas régulier (c'est-à-dire quand tous les sommets sont d'ordre 6). Le masque du schéma est présenté figure 1.2.3 . Un masque constitue une représentation graphique qui indique où les sommets sont insérés (en milieu d'arête pour le butterfly) et comment leurs coordonnées sont calculées, pour le butterfly et avec les notations de la figure 1.2.3 :

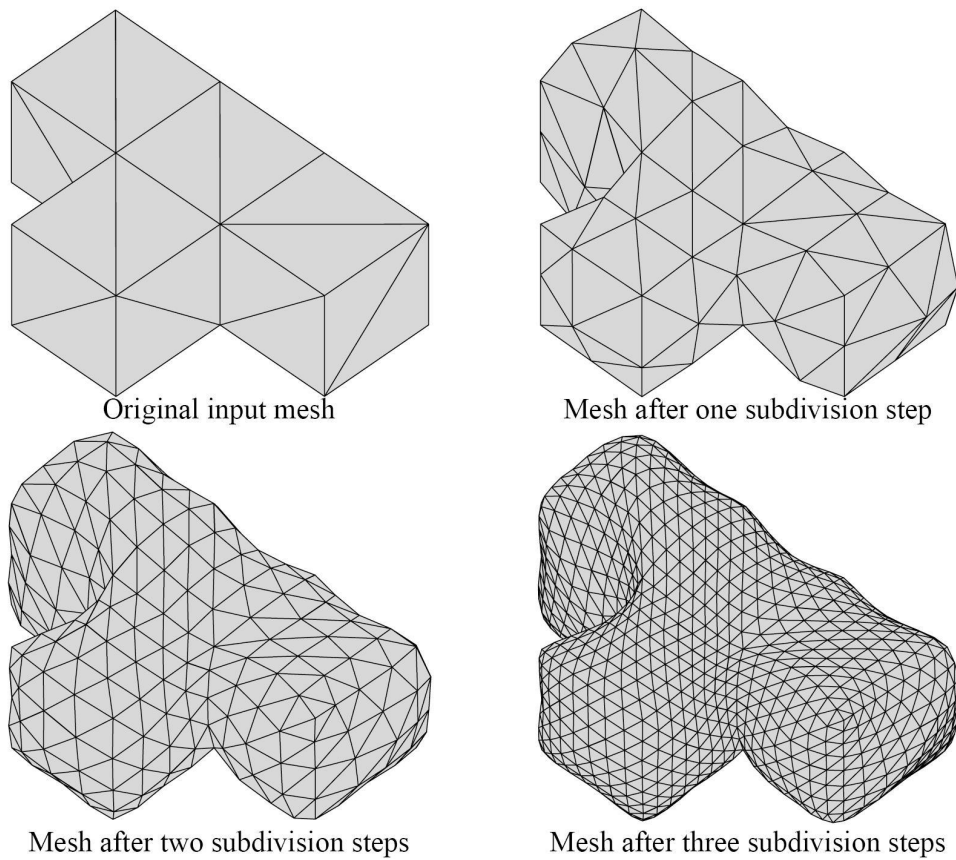
$$d_e = \frac{1}{2}(d_1 + d_2) + 2\omega(d_3 + d_4) - \omega(d_5 + d_6 + d_7 + d_8) \quad (1.2.1)$$

La figure 1.2.4 montre que les surfaces produites, bien que  $C^1$ , ne sont pas aussi lisses que désirées dans le cas où le maillage initial n'est pas régulier. Or c'est le cas de beaucoup de maillages. Pour résoudre ce problème, un pré-traitement ([Alkalai et Dyn, 2004]) peut-être effectué pour générer un maillage régulier. Néanmoins, un tel pré-traitement est non seulement couteux, mais entraîne aussi un changement du maillage initial, ce qui peut être préjudiciable à l'utilisateur. [Zorin *et al.*, 1996] propose donc de modifier le schéma au voisinage des sommets extraordinaires. L'ordre de ces sommets est localement pris en compte selon le masque de la figure 1.2.5 et les équations :

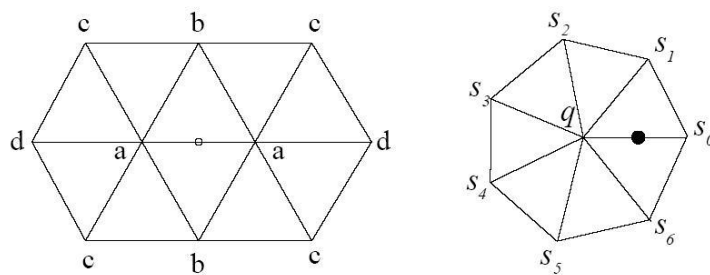
$$\begin{cases} a = \frac{1}{2} - \omega, & b = \frac{1}{8} + 2\omega, & c = -\frac{1}{16} - \omega, & d = \omega \\ s_j = (\frac{1}{4} + \cos(2\pi j/K) + \frac{1}{2} \cos(4\pi j/K))/K \end{cases} \quad (1.2.2)$$

avec  $K$  nombre de voisins du sommet. Ce schéma améliore effectivement les résultats dans le cas de maillages non réguliers (voir figure 1.2.6 ), mais présente l'inconvénient supplémentaire de ne pas être constant. En effet, les coefficients utilisés pour le calcul



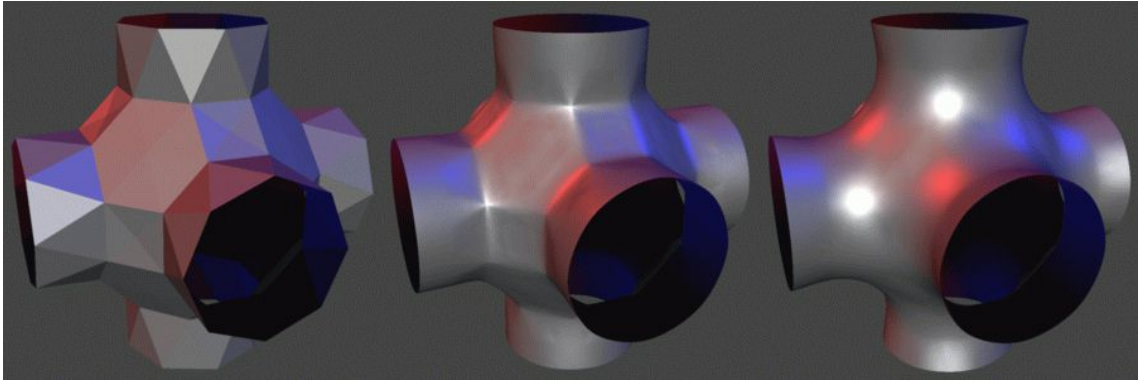


**Fig. 1.2.4** – Exemple de calcul de surface à l'aide du schéma butterfly - Image extraite de [Hubeli et Gross, 2000]



**Fig. 1.2.5** – Masque du schéma butterfly modifié - image extraite de [Zorin *et al.*, 1996]

des nouveaux sommets dépendent du sommet considéré. Par ailleurs, il conserve le défaut principal inhérent aux surfaces de subdivision : l'absence de définition analytique.



**Fig. 1.2.6** – Masque du schéma butterfly modifié - image extraite de [Zorin *et al.*, 1996]

### 1.2.2 Conclusion

Les surfaces de subdivision sont rapides et simples à calculer, les surfaces résultantes sont très lisses en général. Elles sont donc particulièrement adaptées à l'infographie. L'inconvénient majeur qui empêche leur adoption en CFAO est leur manque de définition analytique (certaines expressions ont été trouvées mais restent difficiles à exploiter - [Stam, 1998]).

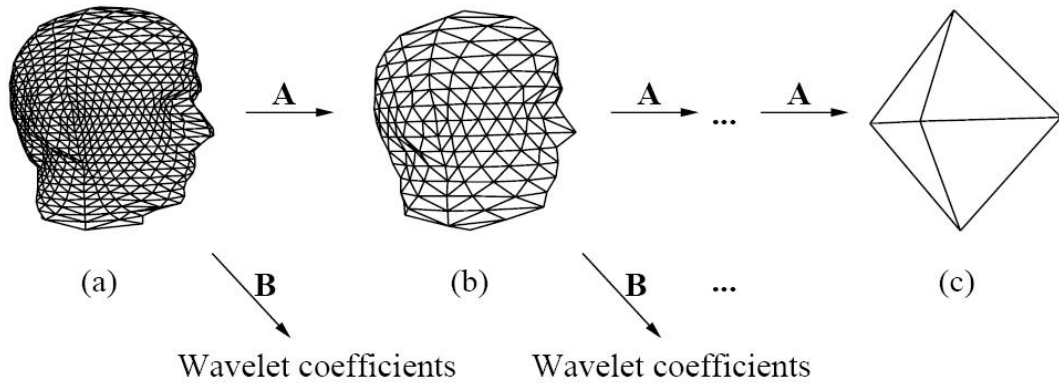
Cette catégorie de surfaces a naturellement donné naissance à la modélisation de surfaces par ondelettes et à l'analyse multirésolution en ne se contentant plus de lisser la surface à chaque étape mais en ajoutant des détails.

## 1.3 Surfaces par ondelettes, Analyse multirésolution

### 1.3.1 Présentation

L'analyse multirésolution découle de l'idée simple de changer de base. Une fonction de  $\mathbb{R}^n$  peut-être représentée dans n'importe quelle base de  $\mathbb{R}^n$ . Néanmoins, certaines s'avèrent plus pratiques que d'autres. Celles utilisées pour l'analyse multirésolution permettent de représenter une fonction comme une fonction simple à basse résolution plus un ensemble de perturbations (ondelettes) de plus en plus fines qui permettent de reconstruire la fonction exactement.

L'analyse multirésolution a été appliquée au traitement du signal dans [Mallat, 1989] et au traitement de l'image dans [Mallat et Hwang, 1992]. Plus récemment, l'analyse



**Fig. 1.3.1** – Décomposition d'un maillage en ondelettes : calcul d'un maillage + des coefficients de détail - Image extraite de [Lounsbery *et al.*, 1997]

multirésolution a été utilisée dans [Finkelstein et Salesin, 1994] pour modéliser des courbes. Grâce à leur utilisation, l'édition de courbe s'effectue aisément de manière globale en conservant les détails locaux.

[Lounsbery *et al.*, 1997] a introduit les ondelettes dans le cas de maillages triangulaires à l'aide d'une subdivision de chaque face en quatre en insérant des sommets aux milieux des arêtes. Cette méthode permet de décrire des surfaces de toute topologie. Grâce aux ondelettes, la compression de maillage et la transmission progressive sont possibles. De même l'édition multirésolution permet de modifier globalement un maillage tout en conservant les détails locaux, propriété particulièrement intéressante. A partir d'un maillage de millions de faces, on dispose de représentations intermédiaires qui le rendent exploitable quel que soit son utilisation. Néanmoins, la contrainte de connectivité compatible avec la subdivision force un travail important de remaillage des modèles existants avant l'analyse multirésolution. Ce travail est présenté dans [Eck *et al.*, 1995].

[Certain *et al.*, 1996] ajoute la couleur aux données et adapte les ondelettes pour gérer ces deux données simultanément, contrairement aux méthodes précédentes qui les séparaient. Une technique de reconstruction est aussi proposée qui prend en compte là-aussi la géométrie et la couleur.

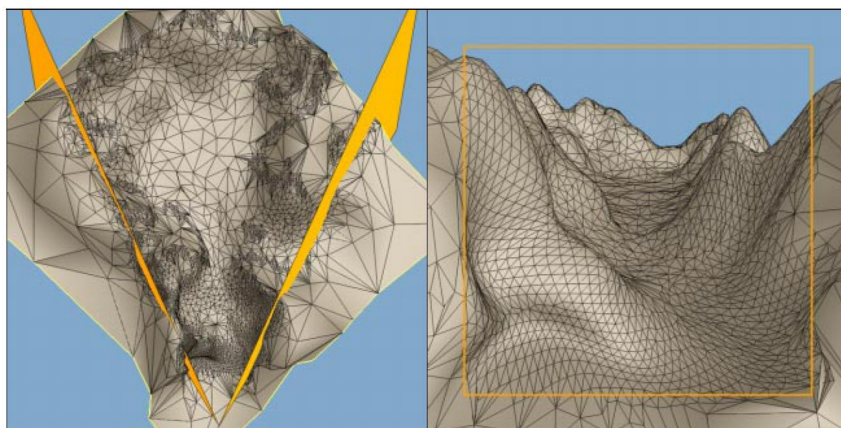
[Valette et Prost, 2004] propose une base multirésolution différente qui prend en compte tous les cas possibles de subdivision. Une face peut ainsi être découpée en une, deux, trois ou quatre faces après raffinement. De cette manière, l'analyse d'un maillage fin existant est simplifiée puisqu'aucune étape de remaillage n'est nécessaire.

### 1.3.2 Conclusion

L'analyse multirésolution propose un cadre mathématique rigoureux à la représentation par niveau de détail. Cela permet d'adapter le niveau de détail affiché en fonction du point de vue, de la fréquence d'affichage désirée ou du débit du réseau. En outre, elle permet de faciliter l'édition car la surface peut être modifiée à n'importe quel niveau de détail. La surface peut être éditée globalement tout en conservant les détails locaux (voir [Finkelstein et Salesin, 1994] et [Gortler et Cohen, 1995] pour une explication détaillée dans le cadre de courbes). Leur mise en pratique est actuellement réduite aux fonctions linéaires, ce qui génère des surfaces linéaires par morceaux : des maillages.

## 1.4 Maillages hiérarchiques

Un maillage possède un nombre fixe de faces. En conséquence, il est soit trop précis si le point de vue est éloigné, soit pas assez précis si, au contraire, on en est trop proche. Une solution consiste à simplifier progressivement un maillage très fin en un maillage grossier de manière à pouvoir choisir la représentation la plus adaptée à l'utilisation. [Rossignac et Borrel, 1993] propose ainsi un algorithme de simplification de maillage adapté principalement à une utilisation en CAO. Cette idée de simplification de maillages est reprise dans [Heckbert et Garland, 1994] en étant adaptée au rendu de scènes complexes. [Hoppe, 1996] et [Hoppe, 1997] étudient la représentation de surface par maillage progressif dans le rendu. Les maillages sont simplifiés par rapport à des critères d'apparence et de temps de rendu.



**Fig. 1.4.1** – Rendu dépendant du point de vue avec les maillages progressifs - Image extraite de [Hoppe, 1996]

## 1.5 Discussion

La modélisation de surface a généré un grand nombre de méthodes de représentation. Toutes ces méthodes sont utilisées couramment dans des domaines précis de l'industrie.

La modélisation basée sur les surfaces polynomiales repose essentiellement sur des méthodes de type produit tensoriel ou basées sur des maillages triangulaires. Les approches de type produit tensoriel sont les plus anciennes et restent les plus répandues grâce aux logiciels de CAO. Elles souffrent néanmoins de leur restriction aux topologies produit tensoriel. Cette restriction est levée par l'utilisation de surfaces à base de maillages triangulaires. Cependant, les calculs en sont plus coûteux. Dans les deux cas, il faut porter attention à leur triangulation en vue de l'affichage. En effet, il est important de trianguler deux surfaces voisines de la même façon pour éviter l'apparition de trous.

Les surfaces de subdivision apportent une solution élégante et rapide aux problèmes de triangulation et d'affichage. En ne manipulant que les maillages successifs, la représentation est consistante à tous les niveaux de précision. De plus leur calcul est rapide car linéaire. Les deux vraies difficultés viennent du fait qu'il est ardu de les rendre interpolantes et surtout de l'absence d'évaluation exacte dans la majorité des cas. Cette catégorie de modélisation est de ce fait réduite à une utilisation en infographie.

La modélisation de surfaces par ondelettes est très prisée de par ses capacités de représentation par niveau de détail. Il est ainsi possible de spécifier la précision de la surface en fonction de paramètres comme la distance au point de vue ou la fréquence d'affichage désirée. De plus, elle offre l'opportunité d'éditer la surface à plusieurs niveaux de détail. Elle permet dès lors de modifier globalement la surface tout en conservant les détails locaux. Néanmoins les surfaces générées ne sont pas lisses (au plan tangent continu) à cause de l'utilisation d'une base de fonctions linéaires par morceaux.

Enfin les maillages hiérarchiques constituent une réponse pragmatique aux problèmes de taille fixe inadaptée des maillages lors de rendu de scènes complexes.

La motivation de notre travail est de trouver un modèle de surface utilisable à la fois en CAO et en infographie. Pour ce faire, il nous est paru évident de dégager plusieurs composantes intéressantes relatives aux différentes surfaces étudiées. En ce qui concerne la CFAO, il est important de travailler avec des surfaces lisses définies paramétriquement. Cela permet une grande souplesse dans leur utilisation aussi bien pour des calculs que pour l'usinage. Cette contrainte rend caduque l'emploi de

surfaces de subdivision (surface limite inconnue, manque de définition analytique), par ondelettes (l'utilisation de fonctions linéaires par morceaux entraînant la création d'une surface linéaire par morceaux) et de maillages hiérarchiques (non lisses). Dans le domaine de la réalité virtuelle, les surfaces à traiter peuvent être de topologie quelconque, avec ou sans bord. Modéliser de telles surfaces est une contrainte importante à laquelle il faut répondre. Cette obligation élimine l'utilisation de surfaces polynomiales de type produit tensoriel. Une méthode polynomiale basée sur un maillage triangulaire constitue donc le seul recours possible pour répondre aux exigences combinées de la CFAO et de l'infographie.

De plus, un principe de niveau de détail est nécessaire pour l'affichage interactif, l'édition hiérarchique comme pour la transmission progressive par réseau. L'unification de représentation de la surface entre les différents niveaux de détails permettrait d'implémenter plus facilement la méthode et minimiser les difficultés de compréhension.

Pour faciliter la procédure de construction d'objet, une méthode interpolante (passant par les sommets de la triangulation) est souhaitable car plus intuitive pour les utilisateurs.

## Conclusion

Dans ce chapitre, les différentes représentations de surface utilisées ont été rappelées ainsi que leurs avantages et inconvénients. Grâce à cela, nous avons pu dresser la liste des propriétés que nous souhaitons obtenir dans la modélisation de surface que nous développons ainsi que les différentes méthodes inadaptées à notre objectif. A partir d'une surface polynomiale basée sur un maillage triangulaire, nous allons donc développer une surface paramétrique lisse (CFAO), interpolante (CFAO et infographie) et hiérarchique (infographie).

Le prochain chapitre détaille la représentation de surface que nous avons créée en prenant en compte les exigences : surface polynomiale définie sur un maillage triangulaire, lisse (continuité du plan tangent), locale, interpolante et surtout hiérarchique (muni de niveaux de détails).



# Chapitre 2

## Interpolation hiérarchique

### Sommaire

---

<b>Introduction</b> . . . . .	<b>34</b>
<b>2.1 Raffinement paramétrique</b> . . . . .	<b>35</b>
<b>2.2 Subdivision et structure de données</b> . . . . .	<b>39</b>
<b>2.3 Interpolation polynomiale</b> . . . . .	<b>42</b>
2.3.1 Continuité du plan tangent . . . . .	43
2.3.2 Continuité du plan tangent aux sommets . . . . .	43
2.3.3 Consistance aux sommets . . . . .	46
2.3.4 Courbes frontières . . . . .	48
2.3.5 Rubans de tangence . . . . .	50
2.3.6 Points libres intérieurs . . . . .	51
2.3.7 Paramètres libres . . . . .	52
2.3.8 Modifications en vue de la hiérarchisation . . . . .	54
2.3.9 Comparaison avec la méthode Butterfly . . . . .	54
<b>2.4 Raffinement local</b> . . . . .	<b>55</b>
Paramètres libres . . . . .	58
Edition locale . . . . .	58
2.4.1 Invariance le long des courbes frontières . . . . .	59
Notations . . . . .	59
Choix des paramètres libres . . . . .	60
Fonctions réelles $\varphi$ , $\mu$ et $\nu$ . . . . .	60
Dérivées secondes (points de contrôle $B_2$ et $B_2^d$ ) . . . . .	62
Courbes de bord . . . . .	64
Plan tangent . . . . .	69



---

2.4.2	Erreur le long des courbes intérieures . . . . .	70
	Dérivées Secondes . . . . .	71
	Courbes de bord . . . . .	72
<b>2.5</b>	<b>Raffinement et édition hiérarchique . . . . .</b>	<b>77</b>
2.5.1	Aspects topologiques . . . . .	77
2.5.2	Aspects géométriques . . . . .	81
<b>2.6</b>	<b>Résultats . . . . .</b>	<b>81</b>
	<b>Conclusion . . . . .</b>	<b>85</b>

---

## Introduction

L'objectif de ce chapitre est de mettre au point une modélisation de surface combinant les avantages des méthodes pré-existantes de type paramétrique et de subdivision. Nous souhaitons ainsi calculer une surface interpolante dite lisse (avec continuité du plant tangent), muni d'une paramétrisation explicite, capable de représenter des surfaces de n'importe quelle topologie et munie d'un système de niveau de détail permettant l'affichage et l'édition multirésolution. Ces conditions assurent la possibilité d'utilisation aussi bien dans le cadre de la CFAO que dans l'infographie.

Pour répondre à ces contraintes, les surfaces sont calculées en plusieurs étapes. Tout d'abord, une première surface simple mais lisse est déterminée à partir d'un maillage initial. Elle est ensuite optimisée à l'aide de paramètres libres de forme. Ce maillage initial est triangulaire et fournit la topologie de la surface, celle-ci pouvant être quelconque, mais limitée à une variété de dimension deux. Ensuite, ce modèle est hiérarchisé de manière à créer et manipuler ces surfaces par niveau de détail. Cette hiérarchisation suit l'idée des H-Splines ([Forsey et Bartels, 1988]) en utilisant des repères locaux pour coder les détails mais est basée sur des maillages triangulaires, ce qui résout les problèmes de topologie. Cette méthode a été publiée dans [Yvart *et al.*, 2004a].

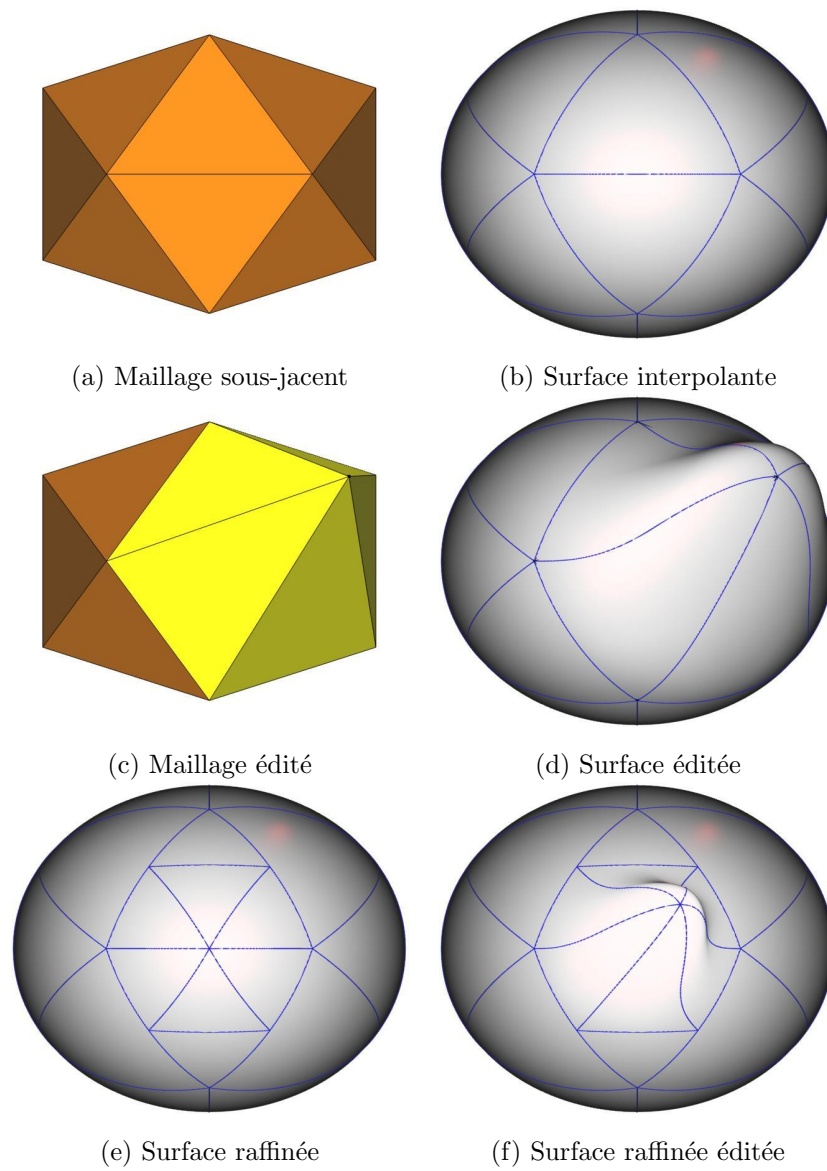
Ce chapitre s'organise comme suit : Section 2.1 présente le problème général de raffinement pour une surface paramétrique. L'opération d'insertion de sommet ainsi que les structures de données utilisées sont ensuite vues section 2.2. La section 2.3 développe le calcul de la surface interpolante, laquelle est raffinée dans la section 2.4. Le principe de l'édition hiérarchique est alors vu au chapitre 2.5.2. Les résultats de notre nouvelle méthode sont alors présentés en chapitre 2.6.

## 2.1 Raffinement paramétrique

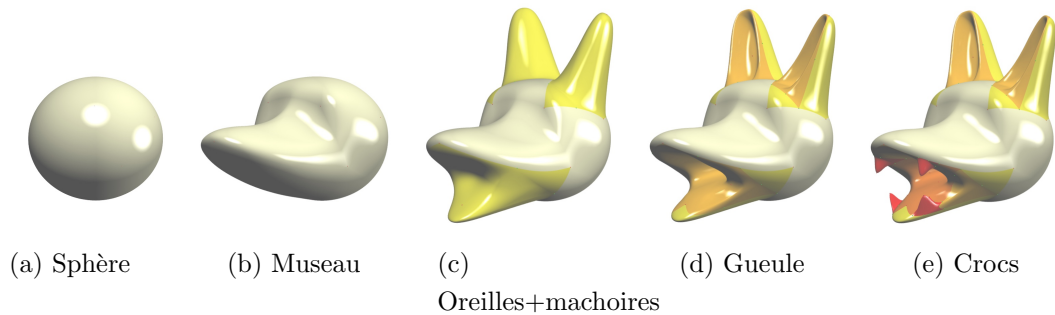
Lorsqu'un objet est représenté par une surface, il est possible de modifier cette surface pour améliorer la représentation de cet objet. Dans le cas de surfaces lisses définies au dessus d'un maillage, cela se fait en modifiant ce maillage ou des paramètres de forme. Les modèles de surfaces lisses définies au dessus d'un maillage sont locaux, c'est-à-dire que le calcul de la surface ne dépend que de la partie du maillage sous-jacent. Cette propriété est particulièrement importante lors de l'édition : lorsqu'un sommet est déplacé, seule la partie de surface autour de ce sommet est modifiée et doit-être recalculée tout en conservant la propriété de continuité  $G^1$ . Cette localité permet donc de faire du design interactif en limitant les calculs à effectuer. Cette technique présente néanmoins un gros défaut : il n'est pas possible d'insérer des détails de taille inférieure à ce support local.

La figure 2.1.1 présente un maillage triangulaire et une surface lisse l'interpolant. On tente d'insérer un détail en déplaçant un sommet du maillage en figure 2.1.1(c). On constate que la modification engendrée correspond à la taille des faces voisines du sommet édité. Si un détail très fin doit-être inséré, cela n'est pas possible car tous les patches voisins du sommet édité sont modifiés. La seule solution serait de construire un maillage initial très fin, de la taille des détails à créer. Néanmoins, cela crée beaucoup de faces inutiles si les détails sont localisés sur une petite partie de la surface. De plus, il est indispensable de connaître la taille la plus petite des détails à générer, et dans le cas de détails plus gros, un ensemble de paramètres sont modifiés pour étendre la modification à la taille voulue. Toutes ces considérations rendent difficile l'utilisation d'un modèle interpolant simple pour modéliser des objets réels, composés d'un aspect globale, et de détails de plus en plus précis. Il est de ce fait indispensable d'introduire une procédure de raffinement pour utiliser les avantages de la multirésolution et représenter un objet d'une manière plus naturelle : une surface initiale donne l'aspect global, puis différents niveaux de détails insèrent les descriptions plus précises de l'objet progressivement. La figure 2.1.2 illustre ce principe avec une tête de chien. Globalement, elle est identifiable à une sphère (figure 2.1.2(a)). Plus précisément, une sphère avec un museau (figure 2.1.2(b)). Le niveau de détails suivant est constitué des deux mâchoires et des oreilles (figure 2.1.2(c)). La gueule et les cavités des oreilles apparaissent ensuite (figure 2.1.2(d)). Viennent enfin les canines (figure 2.1.2(e)). Cette modélisation par niveaux de détails à l'aide de surfaces lisses est le point clé de cette thèse.

Le principe est de découper chaque face en plusieurs autres plus petites. Les sommets sont alors entourés de faces de taille plus petite (voir figure 2.1.1(d)). Lors



**Fig. 2.1.1** – Ajout de détail avant et après raffinement : la taille du détail dépend du niveau de raffinement.



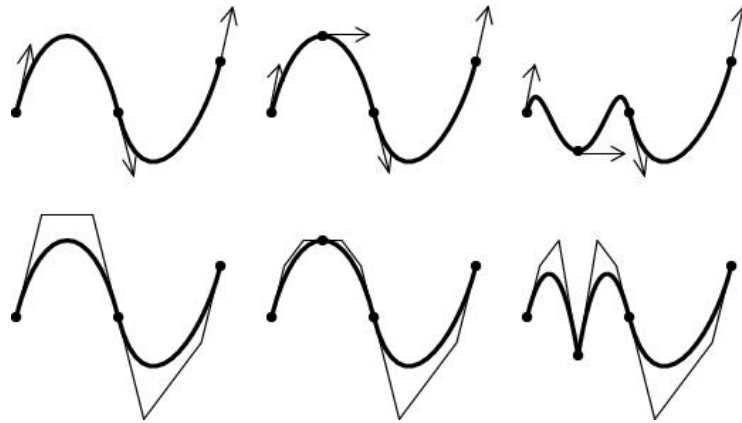
**Fig. 2.1.2** – Principe des niveaux de détails : A partir d’une surface initiale, les détails sont ajoutés en ordre décroissant de leur taille.

de l’édition d’un sommet, seuls les patches voisins sont modifiés, mais ceux-ci sont plus petits (voir figure 2.1.1(e)). Le détail inséré est donc plus petit lui aussi (voir figure 2.1.1). Après un raffinement, des détails plus fins peuvent être ajoutés : cela crée un nouveau niveau de détails. Ces détails ne doivent pas détériorer la continuité de la surface (continuité du plan tangent au minimum). La procédure d’ajout de détail doit prendre en compte cette contrainte.

Dans le cas des surfaces de subdivision, les étapes de raffinement et d’ajout de détails sont aisées. Par nature, un maillage raffiné générera la même surface finale que le maillage initial. Une édition du maillage raffiné implique une modification locale de la surface, le processus de création de la surface par raffinements successifs assure que la surface finale sera lisse.

Pour une surface lisse définie sur un maillage, le maillage initial se comporte comme le maillage de contrôle d’une surface spline. Il contrôle les différentes parties polynomiales de la surface tout en maintenant la continuité entre eux. Avec cette analogie, le raffinement de la surface interpolante est une étape équivalente à l’insertion de nœuds des surfaces splines. De la même façon que l’insertion de nœuds, cette étape introduit une subdivision du domaine paramétrique et du maillage de contrôle sans changer la surface elle-même. Les modifications ultérieures du maillage de contrôle raffiné engendreront des modifications plus locales de la surface puisque le support des nouveaux sommets insérés est plus petit. Néanmoins, la continuité  $G^1$  sera toujours conservée. Pour modifier la surface, l’interpolant paramétrique est appliqué localement sur le nouveau maillage plus fin. De ce fait, il faut que l’interpolant calculé localement donne la même surface que celle avant raffinement. Ainsi, le domaine paramétrique est modifié par le raffinement mais pas la surface en elle-même. Après modification du maillage subdivisé, il faut assurer de conserver la continuité  $G^1$ .

Notre interpolant étant constitué de patches polynomiaux paramétrés sur des triangles sous la forme de patches de Bézier triangulaires, l'idée naturelle pour le subdiviser serait d'utiliser une subdivision de Bézier comme dans [Goldman, 1983]. Cette méthode permet d'obtenir les points de contrôle d'une surface de Bézier après insertion de sommets (voir annexe C). Cette idée, bien qu'intuitive, n'est pas utilisable en pratique puisque toute modification ultérieure des points de contrôle de la surface subdivisée ne conserverait pas nécessairement la continuité globale de la surface. Pour mettre en évidence ce phénomène, intéressons-nous au cas de simples courbes. L'équivalent de notre surface interpolante pour les courbes sont les courbes interpolantes de Hermite. Pour rappel, ces courbes interpolent la position et la dérivée première données en chaque extrémité.



**Fig. 2.1.3** – Le raffinement et l'édition spline conservent la continuité du plan tangent, alors que la subdivision et l'édition de Bézier peuvent détruire cette continuité. Rangée supérieure : courbes de Hermite - raffinement local - édition local qui maintient la continuité  $C^1$ . Rangée inférieure : courbes de Bézier - subdivision de Bézier - déformation locale détruisant la continuité  $G^1$ .

La figure 2.1.3 (à gauche) montre deux parties polynomiales cubiques se joignant avec une continuité  $C^1$ , celle du dessus définie comme une interpolation de Hermite et celle du dessous par un polygone de contrôle de Bézier. La première partie de la courbe est subdivisée (figure 2.1.3 milieu). La figure du haut montre les paramètres de Hermite en résultant (position et dérivées) et celle du bas montre les nouveaux points de contrôle de Bézier. Déplacer un point de contrôle détruit la continuité  $C^1$  (figure 2.1.3 en bas à droite) alors que modifier un paramètre de l'interpolant d'Hermite ne change pas cette continuité (figure 2.1.3 en haut à droite).

Cet exemple simple illustre pourquoi une subdivision de Bézier n'est pas suffisante, et justifie la construction d'un schéma interpolant d'un niveau plus élevé

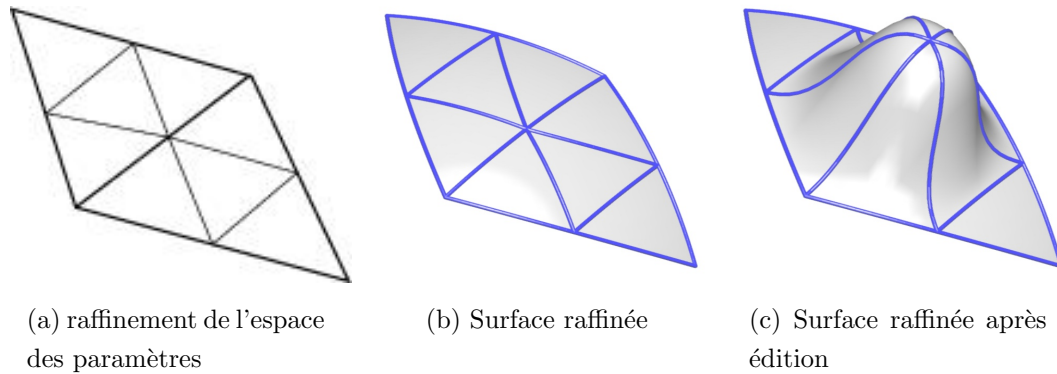
ressemblant aux splines qui constitue l'objet de cet ouvrage.

## 2.2 Subdivision et structure de données

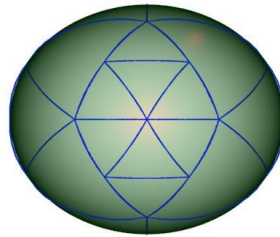
La construction de notre surface hiérarchique suit le schéma suivant : tout d'abord le maillage triangulaire initial est interpolé par une surface lisse. Ensuite, une hiérarchie de surfaces est construite par raffinements locaux successifs de la surface initiale. Pour pouvoir éditer cette surface à n'importe quel niveau de la hiérarchie, ce procédé repose sur une structure de données qui contient à la fois l'information topologique qui encode la subdivision hiérarchique de l'espace des paramètres, et l'information géométrique qui encode la fonction transformant le domaine en la surface.

La surface initiale est calculée sur un maillage triangulaire surfacique. Pour modifier cette surface, il suffit d'éditer les sommets du maillage, c'est-à-dire en changer les coordonnées. Néanmoins, comme vu au chapitre 2.1, ces modifications affectent la surface sur l'ensemble du voisinage de ce sommet, i.e. sur tous les patches de surface partageant ce sommet. Pour traiter des modifications fines, un processus de raffinement est mis en œuvre. De nouveaux sommets sont insérés et les faces raffinées, c'est-à-dire découpées en plusieurs faces filles plus petites. L'édition des nouveaux sommets affectent la surface définie sur leurs faces voisines, qui sont plus petites. La modification est donc plus locale. Pour construire un objet défini sur un maillage, la première chose à faire est donc d'éditer ce maillage pour représenter l'aspect global de l'objet, i.e. les formes représentables avec ce maillage sous-jacent. Après cela, l'espace des paramètres (le maillage) est raffiné localement, et les formes de la taille du maillage raffiné sont ajoutées. On itère alors jusqu'à obtention de tous les détails.

L'opération de base pour raffiner l'espace des paramètres consiste à subdiviser une arête en deux et chaque face adjacente en quatre sous-faces (voir figure 2.2.1 à gauche). Cette opération insère cinq nouveaux sommets dans l'espace des paramètres. Les points de la surface correspondant aux quatre sommets autour dans l'espace des paramètres ne peuvent pas être édités, ils sont dits non-éditables. En effet, ces sommets sont introduits au milieu des arêtes des faces voisines (voir figure 2.2.2). Ils sont fixés géométriquement au milieu de la courbe de bord image de cette arête par construction, et ne peuvent pas bouger. Ils ne sont créés que pour permettre le découpage de la face en quatre faces dites filles et offrir au sommet central un support local. Autour de ce sommet, les faces ne constituent plus une triangulation puisqu'il appartient à une arête. Cependant, cela ne constitue pas un problème car il n'intervient que lors du calcul autour du sommet central. Sa création



**Fig. 2.2.1** – Deux patches adjacents sont subdivisés en quatre sous-patches en insérant un nouveau sommet au milieu de leur arête commune. Ce sommet est un sommet *éditable*. Sa position peut être modifiée, générant une modification locale de la surface tout en conservant la continuité du plan tangent de la surface.

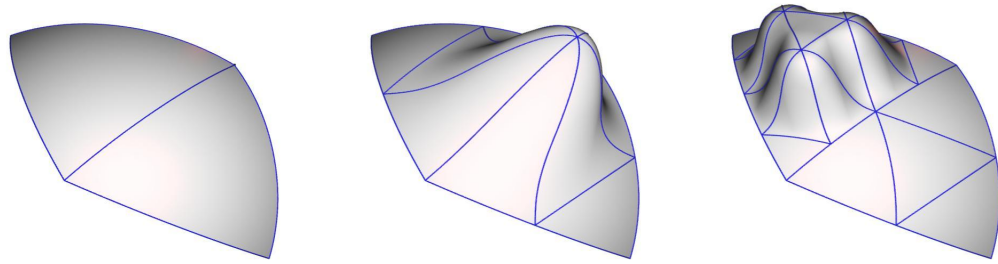


**Fig. 2.2.2** – Subdivision locale d'une surface : deux types de sommets sont créés.

est en fait artificielle. La position du sommet central peut être modifiée librement, ce sommet est dit *éditable* (voir figure 2.2.1 à droite). En effet, localement c'est un sommet d'un maillage régulier puisqu'il dispose de six faces incidentes. L'interpolant est donc reconstruit localement autour de ce sommet (voir section 2.4). La subdivision d'une autre arête peut plus tard entraîner le changement d'état d'un sommet de non-éditable à éditable (voir section 2.5).

Les avantages de ce raffinement en quatre sont multiples. Tout d'abord, il est facile d'ajouter des détails le long des arêtes existantes (voir figure 2.2.3(b)), mais aussi à l'intérieur de chaque face (voir figure 2.2.3(c)). De plus, chaque face est subdivisée en quatre sous-faces de forme similaire. En partant de faces presque équilatérales, on conserve des faces de forme similaire. L'inconvénient de ce découpage résulte du fait que quatre faces sont créées. Le nombre de triangles augmente donc  $\frac{4}{3}$  fois plus rapidement qu'avec une subdivision en trois faces.

Une structure de données adaptée est nécessaire pour pouvoir facilement accéder



(a) Deux faces adjacentes      (b) Détail le long d'un arête      (c) Détail à l'intérieur d'une face

**Fig. 2.2.3** – La subdivision en quatre permet d'insérer des détails aussi facilement le long des arêtes ou à l'intérieur des faces.

aux différents niveaux de détail du haut jusqu'en bas et inversement du bas jusqu'en haut. De la même manière que [Certain *et al.*, 1996], une forêt d'arbres quaternaires est utilisée avec la structure de données suivante :

```
type Vertex = record
    (information géométrique)
    parentFace [2] : tableau de pointeurs vers Face
end record
```

```
type Face = record
    (information géométrique)
    cornerVertex [3] : tableau de pointeurs vers Vertex
    edgeVertex [3] : tableau de pointeurs vers Vertex
    childFace [4] : tableau de pointeurs vers Face
end record
```

Chaque nœud de l'arbre quaternaire encode une face. Les racines sont les faces triangulaires du maillage initial. Quand une face est subdivisée, elle pointe sur ses quatre faces filles *childFaces*. De plus, chaque face pointe vers ses trois sommets (*cornerVertex* dans le champ face record) mais aussi vers ses trois sommets au milieu des arêtes (*edgeVertex* dans le champ face record). Ces trois sommets d'arêtes sont créés dès que la face est construite. Néanmoins ils ne deviennent éditables que lorsque leurs deux faces parentes existent et sont subdivisées. Au niveau 0, les pointeurs *parentFaces* du champ vertex record encodent les relations d'adjacence entre les triangles initiaux.

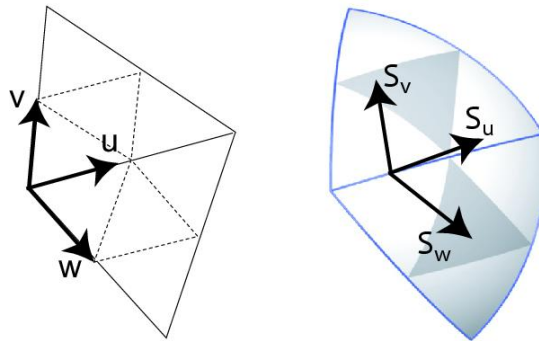


L'utilisation de l'information géométrique se trouve dans la partie 2.4.

## 2.3 Interpolation polynomiale

La construction de notre surface hiérarchique commence en interpolant un maillage initial par une surface de plan tangent continu avant de la raffiner. On impose à cette surface interpolante initiale d'être polynomiale de degré le plus bas possible, et d'être définie localement pour minimiser les calculs à effectuer en cas de modification.

Dans le chapitre 1.1.3, une liste des méthodes qui répondent à ces exigences est donnée. Malheureusement, la plupart sont basées sur la subdivision de Clough-Tocher et ne peuvent donc pas être utilisées dans un contexte de raffinement hiérarchique. En effet, cette subdivision produit des triangles de plus en plus dégénérés après quelques itérations. Après seulement quelques pas de raffinement, les triangles sont aplatis, ce qui mène à un mauvais comportement numérique. De plus la connectivité de chaque sommet est doublée à chaque étape. C'est pourquoi nous choisissons la méthode de [Hahmann et Bonneau, 2003], qui est une extension de [Bonneau et Hahmann, 2000] pour interpoler le maillage initial. Il sera démontré dans la partie 2.4.1 qu'une légère modification est nécessaire pour que cette méthode possède la propriété d'invariance par subdivision le long des courbes de bord, ce qui assure que la surface n'y change pas après une étape de raffinement, et que l'erreur introduite par le raffinement à l'intérieur d'un patch est faible.



**Fig. 2.3.1** – Paramétrisation de l'interpolant. Gauche : Espace des paramètres ; droite : macro-patches avec les directions de tangence.

Dans cette partie, nous rappelons les étapes principales de la construction de cet interpolant. La surface calculée est polynomiale et paramétrée sur le maillage initial. Chaque triangle est envoyé sur un groupe de quatre patches de Bézier de degré cinq. Ce groupe est appelé macro-patch. Quatre sous-triangles coupent le triangle initial

régulièrement comme illustré en figure 2.3.1. Ce découpage en quatre de l'espace des paramètres est la clé qui permet de rendre cet interpolant raffnable, à l'opposé des méthodes vues en section 1.1.3 en tant que méthodes avec subdivision de type Clough-Tocher. La surface construite répond aux contraintes d'être :

- lisse dans le sens de plan tangent continu,
- interpolante,
- polynomiale de degré minimal,
- définie localement et
- hiérarchisable.

La méthode consiste à faire correspondre chaque patch à une face du maillage, chaque courbe de bord correspondant à une arête et interpolant ses extrémités. Chaque macro-patch de la surface est l'image polynomiale par morceau du triangle unité. La construction de la surface découle de l'analyse de la contrainte de continuité du plan tangent dans le cas particulier de surfaces de Bézier regroupées en macro-patches.

### 2.3.1 Continuité du plan tangent

Avec les notations de la figure 2.3.1, la contrainte de continuité du plan tangent entre deux macro-patches s'écrit

$$\varphi(u) \frac{\partial S}{\partial u} = \mu(u) \frac{\partial S}{\partial v} + \nu(u) \frac{\partial S}{\partial w}. \quad (2.3.1)$$

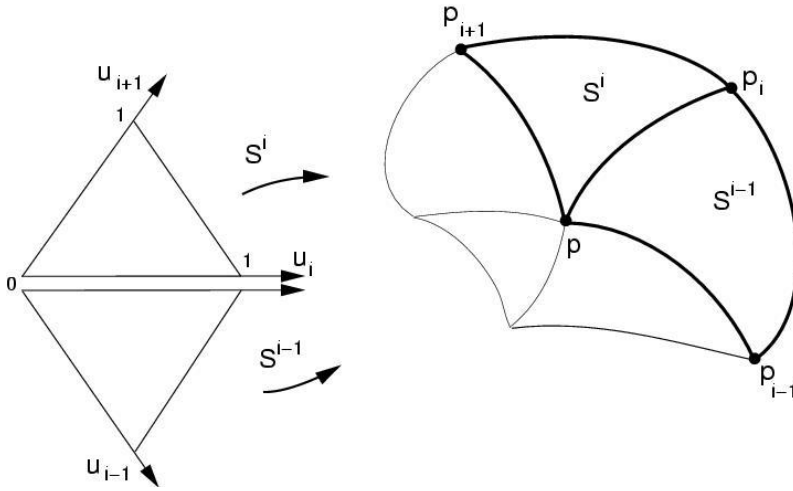
L'équation 2.3.1 dit que les trois dérivées partielles le long de la courbe de bord commune à deux macro-patches sont coplanaires, et donc définissent un plan tangent continu. L'équation 2.3.1 doit être satisfaite le long de toutes les arêtes du maillage initial. Aux sommets du maillage, cela entraîne le problème de consistance au sommet, qui dit que les dérivées secondes et mixtes doivent être choisies avec consistance (voir section 1.1.3).

### 2.3.2 Continuité du plan tangent aux sommets

Le problème de la consistance aux sommets dit que les courbes de bord issues de ce sommet ne peuvent pas être quelconques pour pouvoir construire une surface les interpolant (voir section 1.1.3). Il n'est donc pas possible de construire les courbes de bord indépendamment les unes des autres et il convient d'effectuer un pré-traitement pour autoriser leur interpolation.

Dans le cas présent, ce problème de consistance est mis en évidence en dérivant l'équation (2.3.1) par rapport à  $u$  et en la considérant en 0 :

$$\begin{aligned} \varphi(0) \frac{\partial^2 S}{\partial u^2}(0,0) &= \mu'(0) \frac{\partial S}{\partial v}(0,0) - \varphi'(0) \frac{\partial S}{\partial u}(0,0) + \nu'(0) \frac{\partial S}{\partial w}(0,0) \\ &+ \mu(0) \frac{\partial^2 S}{\partial u \partial w}(0,0) + \nu(0) \frac{\partial^2 S}{\partial u \partial v}(0,0) \end{aligned} \quad (2.3.2)$$



**Fig. 2.3.2** – Notations indexées par le numéro de face

Cette équation est valable pour toutes les courbes issues d'un sommet. Si chaque paramètre est indexé par le numéro de la face considéré comme montré figure 2.3.2, alors :

$$\begin{aligned} \varphi_i(0) \frac{\partial^2 S^i}{\partial u_i^2}(0,0) &= \mu'_i(0) \frac{\partial S^i}{\partial u_{i+1}}(0,0) - \varphi'_i(0) \frac{\partial S^i}{\partial u_i}(0,0) + \nu'_i(0) \frac{\partial S^{i-1}}{\partial u_{i-1}}(0,0) \\ &+ \mu_i(0) \frac{\partial^2 S^{i-1}}{\partial u_i \partial u_{i-1}}(0,0) + \nu_i(0) \frac{\partial^2 S^i}{\partial u_i \partial u_{i+1}}(0,0) \end{aligned} \quad (2.3.3)$$

Pour simplifier l'écriture, on pose :

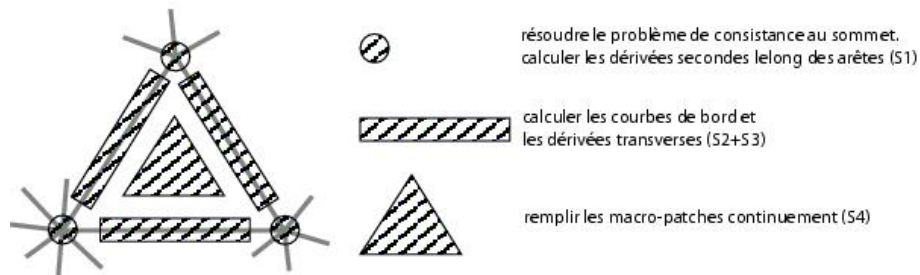
$$D_1^i = \frac{\partial S^i}{\partial u_i}(0,0) \quad (2.3.4)$$

$$D_2^i = \frac{\partial^2 S^i}{\partial u_i^2}(0,0) \quad (2.3.5)$$

$$T^i = \frac{\partial^2 S^i}{\partial u_i \partial u_{i+1}}(0,0) \quad (2.3.6)$$

L'équation (2.3.3) devient :

$$\varphi_i(0) D_2^i = \mu'_i(0) D_1^{i+1} - \varphi'_i(0) D_1^i + \nu'_i(0) D_1^{i-1} + \mu_i(0) T^{i-1} + \nu_i(0) T^i \quad (2.3.7)$$



**Fig. 2.3.3** – Calcul de l'interpolant triangulaire en 4 étapes.

Cette manipulation d'écriture met en évidence l'équation matricielle sous-jacente. En effet, si l'on considère en un sommet le vecteur  $D_1$  de toutes les dérivées premières ( $D_1 = [\dots D_1^i \dots]$ ), le vecteur  $D_2$  de toutes les dérivées secondes ( $D_2 = [\dots D_2^i \dots]$ ) et le vecteur  $T$  de toutes les dérivées croisées ( $T = [\dots T^i \dots]$ ) alors l'équation (2.3.7) se réécrit :

$$M_3.T = M_1.D_1 + M_2.D_2 \quad (2.3.8)$$

Les courbes de bord (dont  $D_1$  et  $D_2$  sont les dérivées premières et secondes) sont donc reliées à un paramètre de la surface. En conséquence, si les courbes de bord sont fixées (ou calculées indépendamment les unes des autres dans une première étape), cela détermine les dérivées secondes croisées de la surface au sommet (appelées twists). Il suffit de résoudre l'équation (2.3.8) en inversant la matrice  $M_3$  pour les obtenir. Dans la pratique, cette matrice  $M_3$  n'est pas inversible. Il n'est donc pas possible de construire une surface interpolant des courbes de bord quelconque. Il faut contraindre le second membre de l'équation (2.3.8) à être dans l'image de la matrice  $M_3$ .

Pour répondre à ces contraintes, l'interpolant est construit en quatre étapes successives (figure 2.3.3) :

- (S1) Le problème de consistance aux sommets est résolu
- (S2) Les courbes de bord sont calculées
- (S3) La continuité du plan tangent entre les macro-patches est satisfaite
- (S4) La continuité à l'intérieur de chaque macro-patch est assurée

Chacune de ces étapes est développée ci-dessous en insistant sur les paramètres libres et leur influence sur l'interpolant. ci-dessous. Pour chaque étape, il ne sera détaillé les équations qu'en une extrémité. Le cas de l'autre extrémité est symétrique.

### 2.3.3 Consistance aux sommets

Le problème de consistance aux sommets vu ci-dessus implique un calcul imbriqué de la surface avec ses courbes de bord. Néanmoins, cette dépendance est minimale puisqu'elle ne concerne que les twists. Il a donc été choisi de fixer les dérivées premières de la courbe et les twists de chaque face. L'équation (2.3.8) est alors validé en l'utilisant pour calculer les dérivées secondes des courbes. Les courbes de bord ne sont plus construites indépendamment mais en fonction de paramètres communs. Il devient alors possible de calculer une surface les interpolant.

Autour d'un sommet d'ordre  $N$ , les  $N$  dérivées premières ainsi que les  $N$  twists (dérivées secondes croisées) des patches qui se rejoignent en ce sommet sont des paramètres libres. A partir de ces paramètres libres, les dérivées secondes des courbes sont calculées de même que les fonctions  $\varphi$ ,  $\mu$  et  $\nu$ .  $\varphi$ ,  $\mu$  et  $\nu$  sont des fonctions scalaires. Ces fonctions réelles sont choisies polynomiales pour assurer à la surface d'être elle-même polynomiale. De plus, la surface générée étant souhaitée de degré minimal, ces fonctions sont fixées linéaires par morceaux, ce qui est le degré minimum. On a donc dans la base de Bézier :

$$\varphi(t) = \varphi_0(1 - t) + \varphi_1 t \quad (2.3.9)$$

$$\mu(t) = \mu_0(1 - t) + \mu_1 t \quad (2.3.10)$$

$$\nu(t) = \nu_0(1 - t) + \nu_1 t \quad (2.3.11)$$

Les valeurs aux extrémités sont obtenues par l'équation (2.3.1). En effet, si les dérivées premières sont fixées, alors l'équation (2.3.1) au point de paramètre (0,0) i.e. au sommet fournit une relation entre les fonctions linéaires et les vecteurs dérivées premières. Pour en déduire les valeurs aux extrémités de  $\varphi$ ,  $\mu$  et  $\nu$ , le produit vectoriel de cette équation avec chacune des dérivées premières est considéré puis projeté sur la normale au plan tangent. Au final :

$$\mu_0 = \frac{\langle n, D_p \wedge D_1 \rangle}{\langle n, D_p \wedge D_n \rangle} \varphi_0 \quad (2.3.12)$$

$$\nu_0 = \frac{\langle n, D_1 \wedge D_n \rangle}{\langle n, D_p \wedge D_n \rangle} \varphi_0, \quad (2.3.13)$$

avec  $n$  vecteur normal à la surface au sommet.  $\varphi_0$  est donc un paramètre libre. Il est fixé tel que  $\mu_0 \cdot \nu_0 = \frac{1}{4}$ .  $\varphi$ ,  $\mu$  et  $\nu$  de l'équation (2.3.1) sont maintenant fixées, et il devient possible de calculer les dérivées secondes des courbes de bord.

Les vecteurs dérivées  $D_1$ ,  $D_n$  et  $D_p$  sont définies comme :

$$D_1 = \frac{\partial S}{\partial u}(0, 0) = D_1^i \tag{2.3.14}$$

$$D_n = \frac{\partial S}{\partial v}(0, 0) = D_1^{i+1} \tag{2.3.15}$$

$$D_p = \frac{\partial S}{\partial w}(0, 0) = D_1^{i-1} \tag{2.3.16}$$

$$\tag{2.3.17}$$

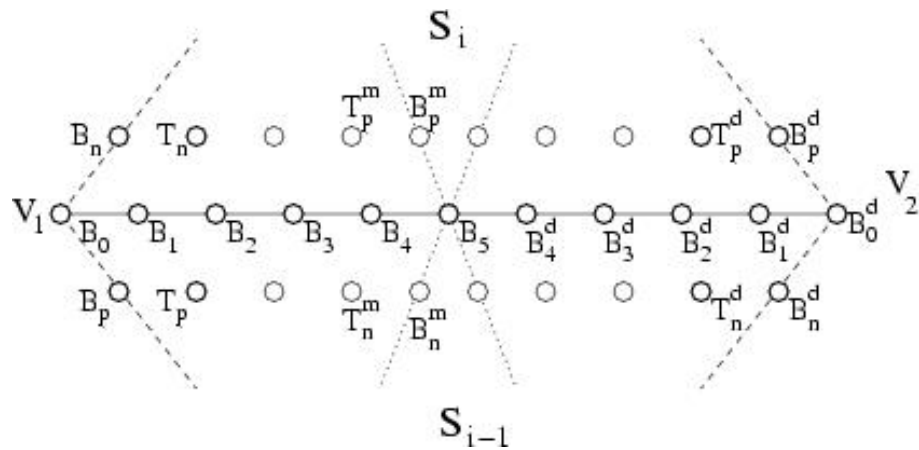
Les dérivées secondes des courbes de bord sont obtenues en dérivant l'équation (2.3.1) et l'évaluant en 0 :

$$\varphi(0)D_2 = \mu'(0)D_p - \varphi'(0)D_1 + \nu'(0)D_n + \mu(0)T_p + \nu(0)T_n \tag{2.3.18}$$

où  $T_n = \frac{\partial^2 S}{\partial u \partial v}(0, 0) = T^i$  et  $T_p = \frac{\partial^2 S}{\partial u \partial w}(0, 0) = T^{i-1}$  sont des vecteurs twists et  $D_2 = \frac{\partial^2 S}{\partial u^2}(0, 0) = D_2^i$  la dérivée seconde de la courbe de bord. Le choix des twists en tant que paramètre libre résout donc bien le problème de consistance au sommet car la dérivée seconde peut toujours être calculée et vérifie bien l'équation (2.3.2).

Cette simple étape de calcul des dérivées secondes à partir des twists résout les problèmes de consistance au sommet.

Un changement dans les paramètres libres (dérivées premières ou twists) affecte les  $N$  macro-patches qui se rejoignent en ce sommet.



**Fig. 2.3.4** – Notations utilisées le long des courbes de bord.

En pratique, la surface calculée est exprimée dans la base de Bézier. Elle est donc contrôlée par des points de contrôle. En terme de points de contrôle de Bézier, en

utilisant les notations de la figure 2.3.4, on a par définition :

$$\begin{aligned} D_1 &= 5(B_1 - B_0) & D_n &= 5(B_n - B_0) \\ D_p &= 5(B_p - B_0) & D_2 &= 20(B_2 - 2B_1 + B_0) \end{aligned}$$

L'équation (2.3.1) transcrite en terme de points de contrôle permet d'identifier terme à terme les points de contrôle. Le coefficient de l'élément  $B_1^5$  de la base de Bézier est  $\frac{4}{5}\varphi_0(B_2 - B_1) + \frac{1}{5}\varphi_1(B_1 - B_0)$  pour le membre de gauche et  $\frac{4}{5}\mu_0(T_p - B_1) + \frac{1}{5}\mu_1(B_p - B_0) + \frac{4}{5}\nu_0(T_n - B_1) + \frac{1}{5}\nu_1(B_n - B_0)$  pour celui de droite. En conséquence,

$$B_2 = \frac{5}{4\varphi_0}A - \frac{\varphi_1}{4\varphi_0}(B_1 - B_0) + B_1 \quad (2.3.19)$$

avec  $A$  valant :

$$A = \frac{4}{5}\mu_0(T_p - B_1) + \frac{1}{5}\mu_1(B_p - B_0) + \frac{4}{5}\nu_0(T_n - B_1) + \frac{1}{5}\nu_1(B_n - B_0) \quad (2.3.20)$$

et permet de calculer le point de contrôle  $B_2$  de la dérivée seconde de la courbe de bord à partir des paramètres libres (twists et dérivées premières). Une fois les points de contrôle de la dérivée seconde fixés, trois points de contrôle à chaque extrémité de la courbe de bord sont définis. La prochaine étape consiste à calculer les points de contrôle restants.

### 2.3.4 Courbes frontières

Les courbes frontières sont calculées de manière à interpoler les dérivés aux extrémités. Les problèmes de consistance aux sommets ont été résolus ci-dessus. La construction de ces courbes de bord est donc relativement libre. Néanmoins, l'équation de continuité  $G^1$  fait intervenir les fonctions scalaires  $\varphi$ ,  $\mu$  et  $\nu$ . Concrètement, les dérivés transverses  $\frac{\partial S}{\partial v}$  et  $\frac{\partial S}{\partial w}$  sont calculés en quotientant la courbe dérivé  $\frac{\partial S}{\partial u}$  par les fonctions scalaires  $\mu$  et  $\nu$ . En prévision de cette division, il convient de mettre ces fonctions scalaires  $\mu$  et  $\nu$  en facteur dans la courbe dérivé  $\frac{\partial S}{\partial u}$ . De cette manière, les dérivés transverses seront polynomiales et non pas rationnelles.

Dans le but de rester polynomial, les courbes frontières sont donc choisies telles que

$$\frac{\partial S}{\partial u} = \mu(u) \nu(u) H(u) \quad \text{le long des courbes frontières,} \quad (2.3.21)$$

avec  $H$  fonction polynomiale de degré 2 par morceaux de continuité  $C^0$ . En effet, l'interpolation des dérivés aux extrémités se fait à travers la construction du polynôme  $H$ . La séparation de  $H$  en deux morceaux raccordés  $C^0$  est introduite pour baisser

le degré total de la surface. Les courbes de bord sont donc des courbes de degré cinq par morceaux, ce qui est le plus bas degré possible vérifiant les contraintes de continuité. Le découpage de la courbe en deux parties amène au découpage de la surface, chaque face correspondant à quatre patches de Bézier (voir figure 2.3.1). Cette subdivision abaisse le degré de la surface mais aussi ouvre la voie à la hiérarchisation de la méthode.

On constate qu'une fois les dérivées au sommet choisies pour résoudre la consistance au sommet, la courbe de bord est entièrement définie. Il n'y a pas de paramètre libre dans cette étape. Soit  $a_0$ ,  $a_1$  et  $a_2$  les points de contrôle du polynôme  $\mu.\nu$  dans la base de Bézier et  $h_0$ ,  $h_1$  et  $h_2$  les points de contrôle du polynôme  $H$  dans la base de Bézier. La construction de  $H$  répond aux équations suivantes :

$$\left\{ \begin{array}{l} h_0 = \frac{5}{a_0}(b_1 - b_0) \\ h_1 = \frac{10}{a_0}(b_2 - b_1) - \frac{a_1}{a_0}h_0 \\ \alpha = \frac{1}{30}(6a_2 + 3a_1 + a_0) \\ \beta = \frac{1}{30}(3a_2h_1 + 4a_1h_1 + a_2h_0) + b_2 \\ h_2 = \frac{\beta^d - \beta}{\alpha + \alpha^d} \end{array} \right. \quad (2.3.22)$$

Ces formules découlent directement de l'interpolation des dérivées aux extrémités et de la continuité demandée. Elles sont obtenues en identifiant terme à terme dans la base de Bézier la formule (2.3.21).

Les points de contrôle restant de la courbe de bord sont obtenus par la même occasion :

$$b_3 = b_2 + \frac{a_0h_2 + 4a_1h_1 + a_2h_0}{30} \quad (2.3.23)$$

$$b_4 = b_3 + \frac{a_1h_2 + a_2h_1}{10} \quad (2.3.24)$$

$$b_5 = b_4 + \frac{a_2h_2}{5} \quad (2.3.25)$$

Tous les points de contrôle de la courbe de bord déterminés, celle-ci est entièrement définie. Un réseau de courbes de bord répondant aux contraintes de consistance aux sommets, et donc interpolables est donc construit. Le long de ces courbes de bord, le plan tangent est maintenant construit.



### 2.3.5 Rubans de tangence

Pour assurer la continuité du plan tangent entre les macro-patches, les dérivées croisées  $\frac{\partial S}{\partial v}$  et  $\frac{\partial S}{\partial w}$  doivent être construites. D'un point de vue de points de contrôle de Bézier, cela revient à calculer la première rangée de points de contrôle de Bézier de chaque côté de la courbe de bord. Ces points sont ceux présentés en figure 2.3.4.

La contrainte de continuité  $G^1$  est réécrite de manière à séparer le traitement de chacun des pathes adjacents :

$$\begin{cases} \frac{\partial S}{\partial v}(u, 0) = \frac{1}{2}(\varphi(u)\nu(u)H(u)) + \nu(u)W(u) \\ \frac{\partial S}{\partial w}(0, u) = \frac{1}{2}(\varphi(u)\mu(u)H(u)) - \mu(u)W(u) \end{cases} \quad (2.3.26)$$

avec  $W$  fonction polynomiale. Cette écriture est équivalente à l'équation (2.3.1). Il suffit pour s'en convaincre d'additionner les deux lignes de l'équation après les avoir multipliées par  $\mu$  et  $\nu$  respectivement.

Le choix de  $W$  détermine le plan tangent. En prime, les points de contrôle de chacun des patches sont calculés séparément après obtention de  $W$ .

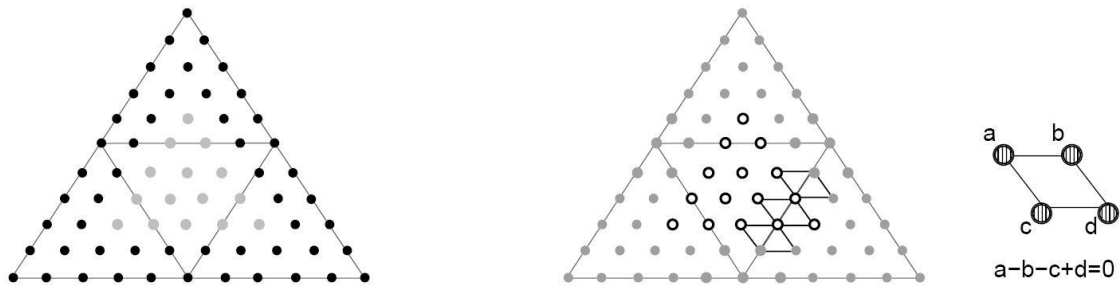
Les dérivées transverses doivent interpoler les valeurs fournies aux extrémités (dérivés transverses et dérivées croisées mixtes). En terme de points de contrôle, les deux points de contrôle de  $W$  à chaque extrémité sont fixé par cette condition d'interpolation.  $W$  peut maintenant être n'importe quel polynôme interpolant ces valeurs. En fait, la contrainte de le garder de degré minimum restreint fortement les choix.  $W$  est soit de degré deux par morceaux raccordés  $C^0$ , soit de degré trois en un seul morceau.

La première équation de 2.3.26 détermine la première rangée de points de contrôle du macro-patch supérieur, la deuxième équation celle du macro-patch inférieur. Il n'y a pas de paramètre libre dans cette étape.

En terme de points de contrôle de Bézier, les points de contrôle  $W_0$  et  $W_1$  de  $W$  dans sa base de Bézier découlent directement de l'identification terme à terme de l'équation (2.3.26) dans la base de Bézier. Le terme en  $B_0^5$  du membre de gauche vaut  $5(B_n - B_0)$  et celui de droite  $\frac{1}{2}\varphi_0\nu_0h_0 + \frac{1}{2}\nu_0W_0$ . De même, le terme en  $B_1^5$  du membre de gauche vaut  $5(T_n - B_1)$  et celui de droite  $\frac{1}{2}(\varphi\nu H)_1 + \frac{1}{2}(\nu W)_1$ , l'indice 1 des polynômes signifiant le terme dans la base  $B_1^5$ . Tous calcul faits :

$$W_0 = \frac{2 \cdot 5}{\nu_0}(B_n - B_0) - \varphi_0 h_0 \quad (2.3.27)$$

$$W_1 = \frac{4 \cdot 5}{\nu_0}(T_n - B_1) - \frac{1}{\nu_0}(\varphi_0\nu_0h_1 + \frac{1}{2}(\varphi_0\nu_1 + \varphi_1\nu_0)h_0 + \frac{1}{2}(\nu_1 + \nu_0)W_0) \quad (2.3.28)$$



**Fig. 2.3.5** – Points de contrôle intervenant dans la continuité intérieure. À droite, règle de parallélogramme.

La construction de la première rangée intérieure des points de contrôle de la surface est obtenue directement de l'écriture de l'équation (2.3.26) dans la base de Bézier après élévation de degré et identification terme à terme.

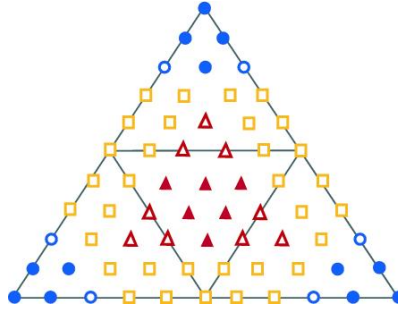
### 2.3.6 Points libres intérieurs

Pour chaque macro-patch, les points de contrôle des courbes de bord et ceux des premières rangées sont fixés. Il reste 15 points de contrôle à déterminer. Ceux-ci sont illustrés en figure 2.3.5. Ces points de contrôle permettent de gérer la continuité de patches de Bézier à l'intérieur des macro-patches. En effet, les quatre patches intérieurs doivent se raccorder de manière au moins  $G^1$  pour obtenir une surface globalement  $G^1$ . En pratique, ils sont raccordés de manière  $C^1$ . La continuité  $C^1$  entre deux patches triangulaires de Bézier revient à contraindre les points de contrôle au voisinage du bord selon la règle dite du parallélogramme, règle illustrée en figure 2.3.5 : Le raccord est  $C^1$  si et seulement si chaque quadruplet de points de contrôle réparti de part et d'autre de la frontière forme un parallélogramme.

En intégrant cette contrainte, six de ces points de contrôle peuvent être choisis librement. Les 9 restants sont calculés de manière à assurer la continuité  $C^1$  entre les 4 patches de Bézier du macro-patch grâce à la règle du parallélogramme. On remarque que les points de contrôle déjà fixés par les étapes précédentes forment bien des parallélogrammes autour des bords intérieurs.

Modifier ces 6 paramètres libres n'affectent qu'un seul macro-patch. La continuité  $C^1$  est obtenue en forçant les points de contrôle répartis le long des courbes de bord intérieures à former des parallélogramme.

La figure 2.3.6 résume à quelle étape chaque point de contrôle de la surface est calculé. Les points représentés par des cercles bleu interviennent dans la résolution



**Fig. 2.3.6** – Points de contrôle d’un macro-patch composé de quatre triangles de Bézier quintiques. Les points de contrôle symbolisés par des cercles sont traités dans l’étape (S1). Les points de contrôle carrés sont calculés en étapes (S2) et (S3). L’étape (S4) implique les points de contrôle représentés par des triangles.

Paramètres libres	Influence
$N$ Dérivées premières et $N$ twists à chaque sommet interpolé de degré $N$	affectent tous les macro-patches autour de ce sommet
6 Points de contrôle intérieurs de chaque macro-patch	affectent un seul macro-patch

**TAB. 2.3.1** – Les paramètres libres contrôlant la surface interpolante et leur influence sur la surface.

du problème de consistance au sommet. Les points modélisés par des carrés jaune sont construits de manière à assurer la continuité  $G^1$  entre les macro-patches tandis que les triangles rouges assurent la continuité  $C^1$  à l’intérieur des macro-patches.

### 2.3.7 Paramètres libres

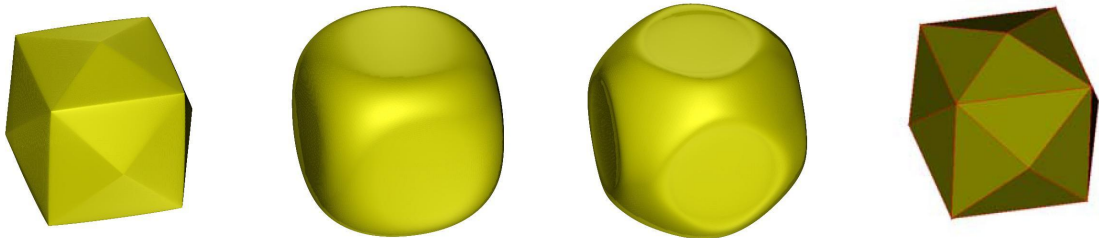
La table 2.3.1 résume les paramètres libres et leur influence sur la surface interpolante. Le champ *information géométrique* des structures *vertex et face records* introduites en section 2.2 est utilisé pour stocker les paramètres libres du schéma d’interpolation.

Ce schéma interpolant présente de nombreux degrés de liberté. Ces multiples paramètres de contrôle sont intéressants pour optimiser la surface générée. Néanmoins, il faut pour cela offrir un moyen simple et intuitif pour définir ces degrés de liberté. Cela se fait grâce à la signification géométrique de chacun de ces paramètres libres.

Tout d’abord, les points extrêmes des patches sont choisis égaux aux sommets

pour interpoler celui-ci. Il convient de remarquer que rien ne l'impose dans la construction de la surface. Ce point de contrôle est théoriquement libre, mais fixé par le choix même d'être interpolant.

En chaque sommet de degré  $n$ , il existe  $n$  dérivées premières libres à déterminer. Ces vecteurs cumulés avec les twists fixent entièrement les courbes de bord. Hors, celles-ci sont primordiales pour l'obtention d'une surface agréable à l'œil. En effet, elles spécifient la direction de départ prise par la courbe de bord. Si celle-ci n'est pas judicieuse, la courbe ondulera pour pouvoir atteindre son extrémité. Il convient donc d'y apporter une attention particulière. Pour cela, la correspondance entre chaque courbe de bord et l'arête sous-jacente est utilisée. Le vecteur dérivée première entre les sommets  $p_1$  et  $p_2$  est obtenu en projetant l'arête entre  $p_1$  et  $p_2$  dans le plan donné par la normale en  $p_1$  selon la direction de la normale en  $p_2$ , le tout pondéré par une constante multiplicative. Cette constante contrôle à quel point la surface est plane au voisinage des sommets (voir figure 2.3.7). Elle représente donc un paramètre intuitif de design. Les normales utilisées sont soit obtenues par moyennage des normales aux faces concourrantes, soit données par l'utilisateur, auquel cas elles constituent un paramètre intuitif supplémentaire.



**Fig. 2.3.7** – Influence de la constante multiplicative des vecteurs dérivées premières sur la surface - image extraite de [Hahmann et Bonneau, 2003].

En chaque sommet d'ordre  $n$ ,  $n$  twists sont à définir. Ils contrôlent les dérivées secondes des courbes de bord (voir équation (2.3.2)) et par là même leur aspect régulier. Des twists nuls sont une idée intéressante, mais pas suffisante pour obtenir une surface satisfaisante. L'absence de règle intuitive rend nécessaire l'utilisation d'une minimisation pour les calculer. De manière plus détaillée, ils sont choisis de manière à minimiser une énergie issue des thinplates :  $\int \frac{\partial^2 S}{\partial u^2} + 2 \frac{\partial^2 S}{\partial u \partial v} + \frac{\partial^2 S}{\partial v^2} dudv$ .

Une fois les dérivées premières et les twists fixés, les courbes de bord sont entièrement déterminées.

Les six points libres intérieurs restant sont calculés eux aussi par une minimisation d'énergie, généralement la même que celle utilisée pour les twists.

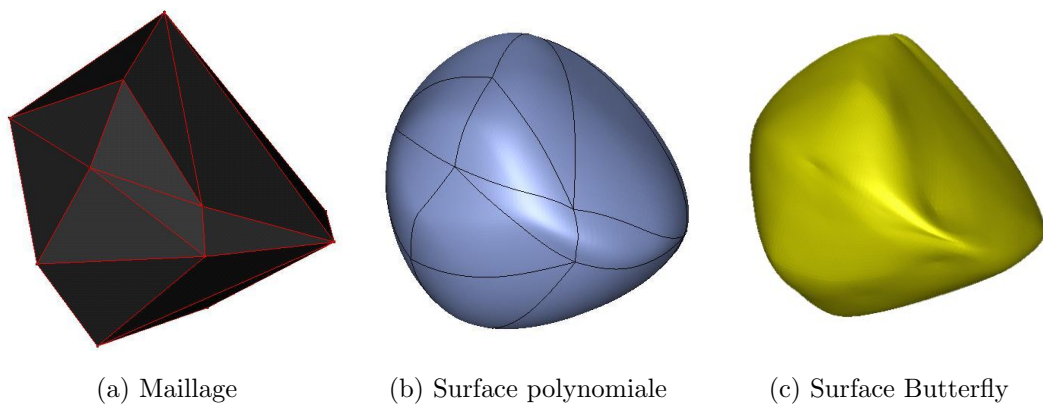
### 2.3.8 Modifications en vue de la hiérarchisation

La méthode issue de [Hahmann et Bonneau, 2003] présenté ci-dessus est une bonne base pour construire une méthode hiérarchique. Cependant, deux modifications sont immédiatement à envisager dans le but de la hiérarchiser.

Tout d'abord, dans [Hahmann et Bonneau, 2003], les fonctions réelles  $\varphi$ ,  $\mu$  et  $\nu$  sont linéaires par morceaux. Il est ainsi pratique de fixer leur valeur à  $\frac{1}{2}$  au point de paramètre  $\frac{1}{2}$ . Cela n'entrave pas le déroulement de l'algorithme ni la qualité des résultats. En vue de la hiérarchisation, il devient nécessaire de les choisir linéaires en un seul morceau. Sans cela, les fonctions après raffinement seraient différentes de celles avant raffinement puisqu'elles prendraient la valeur  $\frac{1}{2}$  trois fois après une étape de raffinement, sept fois après deux étapes et ainsi de suite. Il est clair qu'elles seraient modifiées. L'objectif étant de rendre la surface invariante par raffinement, il convient de choisir ces fonctions linéaires.

De la même façon, [Hahmann et Bonneau, 2003] propose de choisir  $W$  de degré deux par morceaux, avec une continuité  $C^0$ . Dans le but de sa hiérarchisation, il convient de choisir  $W$  cubique en un seul morceau. Cela ne change pas le degré de l'interpolant, mais cela permet d'envisager une étape de subdivision invariante. Ce polynôme défini en un seul morceau, il devient aisé de démontrer qu'il reste inchangé après une étape de subdivision en comparant ses points de contrôle subdivisés avec ceux recalculés. Cette invariance est une étape supplémentaire vers l'invariance globale de la surface.

### 2.3.9 Comparaison avec la méthode Butterfly



**Fig. 2.3.8** – Comparaison de la méthode issue de [Hahmann et Bonneau, 2003] et du butterfly.

La figure 2.3.8 présente les surfaces calculées par la méthode butterfly issue de [Dyn *et al.*, 1990] et celle vue ci-dessus issue de [Hahmann et Bonneau, 2003].

La surface issue de [Hahmann et Bonneau, 2003] est globalement lisse et agréable. Ses courbes de bord correspondent bien aux arêtes du maillage et les tailles respectives des macro-patches correspondent aux tailles respectives des faces du maillage sous-jacent.

La surface issue de [Dyn *et al.*, 1990] présente quant à elle des pincements au voisinage des arêtes séparant des faces de taille fort différente. Ce phénomène est absent des courbes correspondant aux arêtes voisines de deux faces de taille similaire. On observe donc un mauvais comportement de cette méthode dans le cadre de maillage non uniforme, contrairement à la méthode présentée ci dessus.

La réussite de l'interpolant polynomial tient principalement aux nombreux degrés de liberté fournis. En effet, les dérivées premières ont réussi à capter les différences importantes de taille relative des faces du maillage. Au contraire, le schéma butterfly se comporte comme s'il écrasait la surface dans les parties correspondant aux faces de faible aire relative ce qui génère des plis et pincements. La méthode [Hahmann et Bonneau, 2003] est donc avantageuse car en plus d'être polynomiale, les surfaces construites sont aussi plus agréables.

## 2.4 Raffinement local

Une surface étant calculée sur le maillage grossier initial, l'utilisateur peut intégrer et modifier cette surface. Cela peut être fait en éditant l'un des paramètres libres vus en section 2.3.7 ou en éditant l'un des sommets du maillage initial. Néanmoins, ces modifications altéreront toute la surface correspondant à la face ou autour du sommet édité. Cela signifie que seuls des détails de la taille du maillage initial peuvent être ajoutés. Le schéma interpolant ne permet pas d'ajouter des détails de haute fréquence.

Dans cette section, l'étape de base du raffinement présenté en sections 2.1 et 2.2 est détaillée. Ce raffinement résulte du découpage en quatre sous-faces de deux triangles adjacents du domaine. Le sommet inséré au milieu de l'arête commune devient éditable, ce qui signifie que le point correspondant de la surface peut-être spécifié librement par l'utilisateur. Le problème devient alors de construire la surface raffinée à partir du domaine subdivisé de manière à interpoler la position spécifiée par l'utilisateur en ne modifiant que localement la surface et en conservant la continuité du plan tangent globalement.

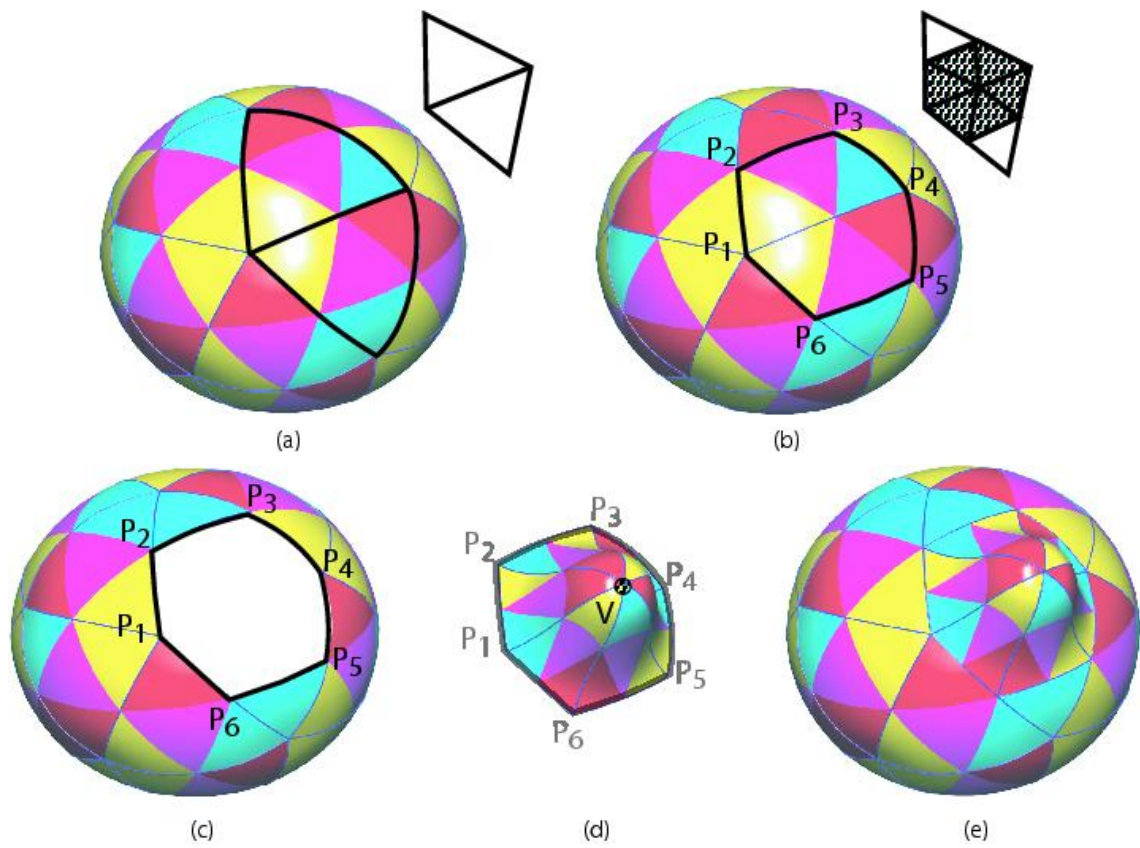
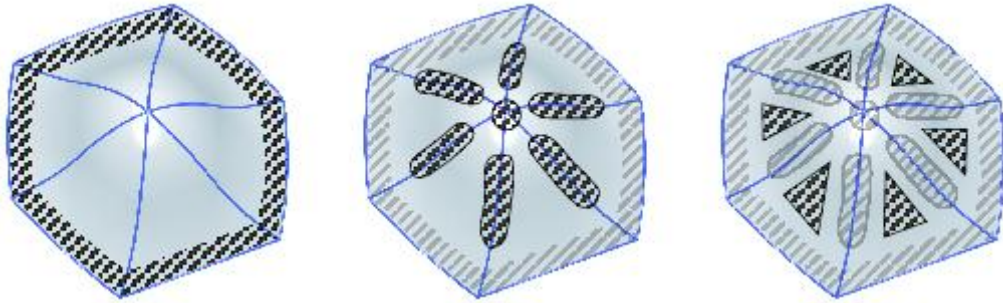


Fig. 2.4.1 – Raffinement local.



**Fig. 2.4.2** – Raffinement local. Gauche : la continuité tangente avec la surface environnante est assurée par une subdivision de Bézier. Milieu et droite : La méthode d’interpolation de la section 2.3 est appliquée localement pour calculer la surface raffinée.

Avant d’aller plus loin, il faut insister sur le fait que cette surface va être construite en utilisant le même schéma d’interpolation que celui utilisé pour calculer la surface initiale en section 2.3. Ci-dessous, les notations de la figure 2.4.1 sont utilisées. Pour des raisons de clarté, cette figure illustre une étape de raffinement à partir de la surface de base, mais la discussion suivante s’applique à des raffinements successifs. La figure 2.4.1(a) montre la surface de base avec des traits gras pour mettre en évidence les deux macro-patches, montrés en haut à droite. Les différentes couleurs correspondent à différents patches de Bézier triangulaires de la surface. Il faut se souvenir que chaque triangle du domaine génère un groupe de quatre patches de Bézier, appelé un macro-patch.

Le procédé de raffinement subdivise les deux triangles du domaine comme montré figure 2.4.1(b) en haut à droite. La partie grisée correspond au morceau de surface entouré par les sommets  $P_1, \dots, P_6$  de la figure 2.4.1(b). Le processus de raffinement remplace cette région composée de 6 patches triangulaires de Bézier par 6 macro-patches, i.e. 24 patches triangulaires de Bézier interpolant le nouveau sommet éditable (voir la figure 2.4.1(d, e) en exemple). En dehors de cette région, la surface n’est pas modifiée. Cela implique en particulier que le nouveau morceau de surface qui est calculé a la même courbe et le même plan tangent le long de ses frontières pour s’intégrer harmonieusement.

Les six nouveaux macro-patches sont obtenus en appliquant le schéma interpolant à un maillage local composé de six triangles autour du sommet intérieur commun, voir figure 2.4.1(d). Ce sommet central  $O$  est choisi librement par l’utilisateur. Les autres sommets  $P_1, \dots, P_6$  sont fixés par la surface autour, voir figure 2.4.1(b).

La construction de l’interpolant qui *remplit le trou* est faite en deux étapes.



Premièrement, une jonction lisse (continuité du plan tangent) avec la surface précédente est établie le long de la courbe de bord en fixant les points de contrôle de la frontière et de la première rangée des macro-patches intérieurs, voir figure 2.4.2(a). Ces points de contrôle sont obtenus à partir des patches de la surface précédente en effectuant une subdivision de Bézier qui découpe chaque patch en quatre sous patches, voir Annexe C. Il faut remarquer qu'en pratique, cette subdivision n'est pas appliquée à tous les patches, mais seulement aux points de contrôle nécessaires.

Deuxièmement, l'interpolant est appliqué. A partir de la surface avant subdivision, des nouveaux paramètres libres pour le calcul de la surface après subdivision sont générés de manière à minimiser les modifications de la surface.

### Paramètres libres

Tout d'abord le nouveau sommet à interpoler est placé au milieu de la courbe de bord subdivisée puisque le schéma est interpolant. Ensuite, les fonctions scalaires  $\varphi$ ,  $\mu$  et  $\nu$  dépendent d'un paramètre libre  $\varphi(0)$  (voir l'équation (2.3.12)). Au sommet nouvellement inséré, on le choisit comme la valeur de  $\varphi(\frac{1}{2})$ . En effet, ces fonctions sont linéaires. Pour que le raffinement, à savoir un découpage en deux parties égales, ne les changent pas, il suffit de fixer comme valeur au milieu la valeur réelle en  $\frac{1}{2}$ . Les dérivées premières et mixtes sont choisies en prenant leur valeur au point milieu de la courbe de bord subdivisée. Les points libres intérieurs sont eux calculés avec une subdivision de type Goldman (voir annexe C). De telles choix des paramètres libres permet à la surface de n'être pratiquement pas modifiée. Leur valeur exacte ainsi que leur justification sont expliquées dans la section 2.4.1.

Une fois que les paramètres libres relatifs au sommet intérieur et aux faces intérieures sont obtenus, l'interpolant recalcule la surface selon l'algorithme de la section 2.3 : le problème de consistance au sommet est résolu pour le sommet intérieur, les courbes de bord et les dérivées croisées trans-frontières sont calculées, et enfin les macro-patches sont remplis.

### Edition locale

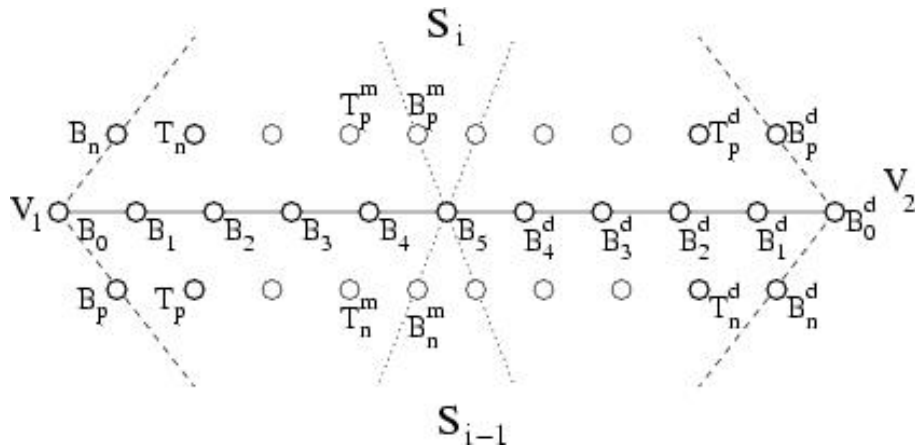
L'utilisateur peut alors éditer la surface localement. En effet, au voisinage du sommet inséré, la surface est définie sur un maillage à l'aide de ses paramètres libres. Plusieurs choix s'offrent à lui pour modifier la surface. Tout d'abord, il lui est possible de spécifier la position du sommet nouvellement créé. Il édite ainsi le maillage fin obtenu après raffinement. Il peut aussi mettre à profit les paramètres libres offerts par l'interpolant. Dans tous les cas, il insère ainsi des détails locaux

puisque leur support est plus fin qu'avant la subdivision.

### 2.4.1 Invariance le long des courbes frontières

Lorsque la surface est recalculée après subdivision pour remplacer les patches du niveau précédent, on souhaite qu'elle ne soit pas modifiée. Pour cela, il faut choisir judicieusement les paramètres libres et étudier les conséquences d'un raffinement sur la surface. Tout d'abord, la surface le long de la courbe de bord qui vient d'être raffinée est étudiée pour montrer qu'elle n'est pas modifiée. Ensuite, l'intérieur du patch est considéré. Cette analyse a été présentée dans [Hahmann *et al.*, 2003].

#### Notations



**Fig. 2.4.3** – Notations utilisée le long des courbes de bord.

Les notations suivantes sont utilisées (voir figure 2.4.3) :  $S$  désigne la surface,  $S_{i-1}$  et  $S_i$  les deux patches de surface partageant une courbe de bord commune  $\Gamma^R$ . Les paramètres libres aux extrémités sont assimilés à leur point de contrôle : respectivement  $B_1, B_p, B_n, T_p$  et  $T_n$  pour le sommet  $V_1$  et  $B_1^d, B_p^d, B_n^d, T_p^d$  et  $T_n^d$  pour le sommet  $V_2$  (à droite sur la figure). Les points de contrôle de la première moitié de la courbe de bord sont les  $B_0$  (extrémité),  $B_1$  (point de la dérivée première) etc jusqu'à  $B_5$ . La seconde est contrôlée par les  $B_i^d$ , pour  $i = 0, \dots, 5$ . Les points  $B_n^m, B_p^m, T_n^m$  et  $T_p^m$  sont les points de contrôle de la surface spécifiés par la figure (m pour milieu).

$\varphi^R, \mu^R$  et  $\nu^R$  sont les fonctions réelles linéaires décrites dans l'équation (2.3.1) et valant  $\varphi_0^R, \mu_0^R$  et  $\nu_0^R$  en  $V_1$  et  $\varphi_1^R, \mu_1^R$  et  $\nu_1^R$  en  $V_2$  (le R en exposant est utilisé pour les valeurs avant raffinement).

Après un raffinement,  $\Gamma^R$  est scindé en deux courbes  $\Gamma_1$  et  $\Gamma_2$ . Seul le cas de  $\Gamma_1$  est détaillé, une simple symétrie permettant d'obtenir les équations pour  $\Gamma_2$ . Les paramètres libres aux extrémités sont de nouveau assimilés à leur point de contrôle :  $b_1, b_p, b_n, t_p$  et  $t_n$  pour le sommet  $V_1$  et les  $b_1^d, b_p^d, b_n^d, t_p^d$  et  $t_n^d$  pour le sommet inséré  $v$ . Les points de contrôle de la première moitié de la courbe de bord sont les  $b_0$  (extrémité),  $b_1$  (point de la dérivée première) etc jusqu'à  $b_5$ . La seconde est contrôlée par les  $b_i^d$ , pour  $i = 0, \dots, 5$  (de la même façon que pour la surface avant raffinement mais en minuscule).  $\varphi, \mu$  et  $\nu$  sont les fonctions réelles linéaires mentionnées dans l'équation (2.3.1) et valant  $\varphi_0, \mu_0$  et  $\nu_0$  en  $V_1$  et  $\varphi_1, \mu_1$  et  $\nu_1$  en  $v$ .

### Choix des paramètres libres

Lors d'un raffinement, les nouveaux paramètres libres (toujours assimilés à leur point de contrôle) sont calculés en fonction de la surface au niveau précédent. Ils sont choisis de manière à définir les mêmes positions et dérivées au milieu de la courbe de bord qui a été raffinée :

$$\left\{ \begin{array}{ll} b_0 = B_0 & b_0^d = B_5 \\ b_1 = \frac{B_0 + B_1}{2} & b_1^d = \frac{B_5 + B_4}{2} \\ b_n = \frac{B_0 + B_n}{2} & b_n^d = \frac{B_5 + B_n}{2} \\ b_p = \frac{B_0 + B_p}{2} & b_p^d = \frac{B_5 + B_p}{2} \\ t_n = \frac{B_0 + B_1 + B_n + T_n}{4} & t_n^d = \frac{B_5 + B_1 + B_n + T_n^m}{4} \\ t_p = \frac{B_0 + B_1 + B_p + T_p}{4} & t_p^d = \frac{B_5 + B_1 + B_p + T_p^m}{4} \end{array} \right. \quad (2.4.1)$$

Ces paramètres libres permettent de calculer la surface raffinée. On va montrer que les courbes de bord ainsi que le plan tangent défini le long de celles-ci restent inchangés après une étape de raffinement. A l'aide de ces notations, on va d'abord montrer que  $\Gamma_1(u) = \Gamma^R(\frac{u}{2})$  pour  $u \in [0, 1]$ , c'est-à-dire que les courbes de bord ne subissent qu'un changement de paramétrisation.

### Fonctions réelles $\varphi, \mu$ et $\nu$

Les courbes de bord sont définies par les paramètres libres en passant par plusieurs étapes intermédiaires. Tout d'abord, les fonctions réelles  $\varphi, \mu$  et  $\nu$  sont calculées par la détermination de leur valeur aux extrémités. Soient  $d_i$  les différentes

dérivées aux extrémités (à un facteur près) :

$$\begin{array}{ll}
d_1 = b_1 - b_0 & d_1^d = b_1^d - b_0^d \\
d_n = b_n - b_0 & d_n^d = b_n^d - b_0^d \\
d_p = b_p - b_0 & d_p^d = b_p^d - b_0^d \\
d_2 = b_2 - 2b_1 + b_0 & d_2^d = b_2^d - 2b_1^d + b_0^d \\
D_1 = B_1 - B_0 & D_n = B_n - B_0 \\
D_p = B_p - B_0 & D_2 = B_2 - 2B_1 + B_0 \\
D_1^m = B_4 - B_5 & \\
D_p^m = B_p^m - B_5 & D_n^m = B_n^m - B_5
\end{array}$$

Pour la construction de l'interpolant, des fonctions réelles  $\varphi$ ,  $\mu$  et  $\nu$  interviennent. La première étape est de montrer que ces fonctions sont inchangées par le raffinement. Nous calculons donc les nouvelles fonctions à l'aide de celles avant raffinement. Avec les notations ci-dessus, on a par construction (équations (2.3.12) et (2.3.13) de l'interpolant) :

$$\begin{aligned}
\mu_0 &= \frac{\langle n, d_p \wedge d_1 \rangle}{\langle n, d_p \wedge d_n \rangle} \varphi_0 \\
\nu_0 &= \frac{\langle n, d_1 \wedge d_n \rangle}{\langle n, d_p \wedge d_n \rangle} \varphi_0
\end{aligned}$$

où  $n$  est le vecteur unitaire normal au plan contenant les  $d_i$ .

$\varphi_0$  et  $\varphi_1$  sont des paramètres libres. Ils sont choisis à partir des fonctions réelles avant raffinement par :

$$\varphi_0 = \varphi_0^R \tag{2.4.2}$$

$$\varphi_1 = \frac{\varphi_0^R + \varphi_1^R}{2} \tag{2.4.3}$$

Ce choix permet de rendre ces fonctions invariantes par subdivision, ce qui est prouvé ci-après.

On va commencer par montrer que  $\mu(u) = \mu^R(\frac{1}{2}u)$ ,  $\nu(u) = \nu^R(\frac{1}{2}u)$  et  $\varphi(u) = \varphi^R(\frac{1}{2}u)$ , i.e. que les fonctions  $\varphi$ ,  $\mu$  et  $\nu$  restent inchangées.

On a :

$$\varphi_0 = \varphi_0^R \text{ fixé par l'équation (2.4.2) pour obtenir l'invariance}$$

$$\begin{aligned}
\varphi_1 &= \frac{\varphi_0^R + \varphi_1^R}{2} \text{ fixé par l'équation (2.4.3) pour obtenir l'invariance} \\
&= \varphi^R\left(\frac{1}{2}\right)
\end{aligned}$$

avec  $\varphi(t) = \varphi_0(1-t) + \varphi_1 t$  (donné par l'interpolant dans l'équation (2.3.9)) linéaire. On en déduit  $\varphi(u) = \varphi^R(\frac{1}{2}u)$ .

Pour  $\mu$  :

$$\begin{aligned}\mu_0 &= \frac{\langle n, d_p \wedge d_1 \rangle}{\langle n, d_p \wedge d_n \rangle} \varphi_0 \text{ d'après (2.3.12)} \\ &= \frac{\langle n, \frac{1}{2}D_p \wedge \frac{1}{2}D_1 \rangle}{\langle n, \frac{1}{2}D_p \wedge \frac{1}{2}D_n \rangle} \varphi_0^R \\ &= \frac{\langle n, D_p \wedge D_1 \rangle}{\langle n, D_p \wedge D_n \rangle} \varphi_0^R \\ &= \mu_0^R\end{aligned}$$

( $n$  est le vecteur unitaire normal au plan des  $d_i$  et  $D_i$ ).

Avec la même notation, i.e.  $n^m$  est le vecteur unitaire normal aux  $D_i^m$  :

$$\begin{aligned}\mu_1 &= \frac{\langle n^m, d_p^d \wedge d_1^d \rangle}{\langle n, d_p^d \wedge d_n^d \rangle} \varphi_1 \\ &= \frac{\langle n^m, \frac{1}{2}D_p^m \wedge \frac{1}{2}D_1^m \rangle}{\langle n^m, \frac{1}{2}D_p^m \wedge \frac{1}{2}D_n^m \rangle} \frac{\varphi_0^R + \varphi_1^R}{2} \\ &= \frac{\mu_R(\frac{1}{2}) \varphi_0^R + \varphi_1^R}{\varphi^R(\frac{1}{2}) 2} \\ &= \mu^R(\frac{1}{2}) \\ &= \frac{\mu_0^R + \mu_1^R}{2}\end{aligned}$$

car  $\mu_R(\frac{1}{2}) = \frac{\langle n^m, \frac{1}{2}D_p^m \wedge \frac{1}{2}D_1^m \rangle}{\langle n^m, \frac{1}{2}D_p^m \wedge \frac{1}{2}D_n^m \rangle} \varphi^R(\frac{1}{2})$  vu que l'équation de continuité  $G^1$  est vérifiée en  $\frac{1}{2}$  et  $\varphi^R(\frac{1}{2}) = \frac{\varphi_0^R + \varphi_1^R}{2}$  (puisque  $\varphi$  est linéaire). On a donc  $\mu_0 = \mu_0^R$  et  $\mu_1 = \mu^R(\frac{1}{2})$ . Comme  $\mu(t) = \mu_0(1-t) + \mu_1 t$  (d'après (2.3.10)) est une fonction linéaire on en déduit  $\mu(u) = \mu^R(\frac{1}{2}u)$ . On démontre de la même façon que  $\nu(u) = \nu^R(\frac{1}{2}u)$ .

Nous avons donc démontré que les nouvelles fonctions réelles  $\varphi$ ,  $\mu$  et  $\nu$  sont une reparamétrisation de celles avant raffinement :  $\mu(u) = \mu^R(\frac{1}{2}u)$ ,  $\nu(u) = \nu^R(\frac{1}{2}u)$  et  $\varphi(u) = \varphi^R(\frac{1}{2}u)$  et restent donc inchangées.

### Dérivées secondes (points de contrôle $B_2$ et $B_2^d$ )

La deuxième étape du calcul de l'interpolant implique la détermination des dérivées secondes à partir des paramètres libres et des fonctions réelles. On montre donc que les points de contrôle de la dérivée seconde qui sont calculés par l'interpolant coïncident avec ceux obtenus par subdivision (i.e. que  $b_2 = \frac{1}{4}B_2 + \frac{1}{2}B_1 + \frac{1}{4}B_0$ ).

En  $V_1$  on a par construction de l'interpolant (équations (2.3.19) et (2.3.20)) :

$$B_2 = \frac{5}{4\varphi_0^R} A^R - \frac{\varphi_1^R}{4\varphi_0^R} (B_1 - B_0) + B_1 \quad (2.4.4)$$

$$b_2 = \frac{5}{4\varphi_0} A - \frac{\varphi_1}{4\varphi_0} (b_1 - b_0) + b_1 \quad (2.4.5)$$

avec  $A$  et  $A^R$  valant :

$$A^R = \frac{4}{5}\mu_0^R(T_p - B_1) + \frac{1}{5}\mu_1^R(B_p - B_0) + \frac{4}{5}\nu_0^R(T_n - B_1) + \frac{1}{5}\nu_1^R(B_n - B_0) \quad (2.4.6)$$

$$A = \frac{4}{5}\mu_0(t_p - b_1) + \frac{1}{5}\mu_1(b_p - b_0) + \frac{4}{5}\nu_0(t_n - b_1) + \frac{1}{5}\nu_1(b_n - b_0) \quad (2.4.7)$$

On va montrer que  $b_2 = \frac{B_0 + 2B_1 + B_2}{4}$  :

$$\begin{aligned} A &= \frac{4}{5}\mu_0(t_p - b_1) + \frac{1}{5}\mu_1(b_p - b_0) + \frac{4}{5}\nu_0(t_n - b_1) + \frac{1}{5}\nu_1(b_n - b_0) \text{ d'après (2.4.1)} \\ &= \frac{4}{5}\mu_0^R \left( \frac{T_p + B_p + B_0 + B_1}{4} - \frac{B_0 + B_1}{2} \right) + \frac{1}{5} \frac{\mu_0^R + \mu_1^R}{2} \left( \frac{B_0 + B_p}{2} - B_0 \right) \\ &\quad + \frac{4}{5}\nu_0^R \left( \frac{T_n + B_n + B_0 + B_1}{4} - \frac{B_0 + B_1}{2} \right) + \frac{1}{5} \frac{\nu_0^R + \nu_1^R}{2} \left( \frac{B_0 + B_n}{2} - B_0 \right) \\ &= \frac{1}{4}A^R + \frac{1}{4}\mu_0^R(B_p - B_0) + \frac{1}{4}\nu_0^R(B_n - B_0) \text{ d'après (2.4.6)} \\ &= \frac{1}{4}A^R + \frac{1}{4}\varphi_0^R(B_1 - B_0) \text{ vu (2.3.1)} \end{aligned} \quad (2.4.8)$$

On a donc :

$$\begin{aligned} b_2 &= \frac{5}{4\varphi_0} A - \frac{\varphi_1}{4\varphi_0} (b_1 - b_0) + b_1 \\ &= \frac{5}{4\varphi_0^R} \left[ \frac{1}{4}A^R + \frac{1}{4}\varphi_0^R(B_1 - B_0) \right] - \frac{\varphi_0^R + \varphi_1^R}{2 \cdot 4\varphi_0^R} \left( \frac{B_0 + B_1}{2} - B_0 \right) \\ &\quad + \frac{B_0 + B_1}{2} \text{ vu (2.4.1) et (2.4.8)} \\ &= \frac{1}{4} \left[ \frac{5}{4\varphi_0^R} A^R - \frac{\varphi_1^R}{4\varphi_0^R} (B_1 - B_0) + B_1 \right] + \frac{1}{4} \left[ \frac{\varphi_1^R}{4\varphi_0^R} (B_1 - B_0) + B_1 \right] \\ &\quad + \frac{5}{4\varphi_0^R} \frac{1}{4} \varphi_0^R (B_1 - B_0) - \frac{\varphi_0^R + \varphi_1^R}{2 \cdot 4\varphi_0^R} \left( \frac{B_0 + B_1}{2} - B_0 \right) + \frac{B_0 + B_1}{2} \\ &= \frac{1}{4}B_2 + \frac{1}{2}B_1 + \frac{1}{4}B_0 \end{aligned} \quad (2.4.10)$$

La dérivée seconde construite en  $V_1$  par l'interpolant est donc bien la même que celle obtenue après subdivision. Ces calculs ne présentaient pas de difficultés dans la mesure où  $B_2$  était calculé par la même méthode que  $b_2$ . Le problème se pose

en  $v$  car  $b_2^d$  est calculé par l'interpolant avec la formule vue ci-dessus alors que  $B_3$  est calculé autrement. La seule chose dont nous disposons est l'information selon laquelle la surface est  $G^1$  partout, en particulier au milieu de la courbe de bord. Or l'équation définissant  $B_2$  et  $b_2$  découle directement de l'équation de continuité  $G^1$ . On en déduit que  $B_3$  la respecte et que donc :

$$B_3 = \frac{5}{4\varphi_m^R} A_m^R - \frac{\varphi_0^R}{4\varphi_m^R} (B_4 - B_5) + B_4 \text{ vu (2.3.19) au milieu de la courbe de bord}$$

$$A_m^R = \frac{4}{5} \mu_m^R (T_p^m - B_4) + \frac{1}{5} \mu_0^R (B_p^m - B_5) + \frac{4}{5} \nu_m^R (T_n^m - B_4)$$

$$+ \frac{1}{5} \nu_0^R (B_n^m - B_5) \text{ vu (2.3.20) au milieu de la courbe de bord}$$

(Cela se démontre en considérant l'équation de continuité  $G^1$  en  $\frac{1}{2}$  puis en comparant les points de contrôle de chacun des membres.) Une fois cette propriété acquise, on démontre de la même façon que pour  $b_2$  que  $b_2^d = \frac{1}{4}B_3 + \frac{1}{2}B_4 + \frac{1}{4}B_5$  c'est-à-dire que le point de contrôle  $b_2^d$  est le même que celui obtenu par subdivision.

### Courbes de bord

Les trois premiers points de chaque extrémité reconstruits par l'interpolant sont identiques à ceux obtenus par subdivision. Il reste à montrer que les 5 autres vérifient cette propriété. En fait, vu que la courbe est contruite de deux morceaux, on peut se contenter de démontrer cette propriété pour 3 points :  $b_3$ ,  $b_4$  et  $b_5$ . Ceux-ci sont construits à l'aide d'une fonction  $H$  composée de deux polynômes de degré 2 raccordés  $C^0$  définie comme expliqué dans la section 2.3 par l'équation (2.3.21) :

$$\Gamma'_1(u) = \mu(u)\nu(u)H(u).$$

Si on écrit les polynômes  $\mu \cdot \nu$  et  $H$  dans la base de Bézier, ils sont définis par leurs points de contrôle respectivement  $a_0, a_1, a_2$  et  $h_0, h_1, h_2$ .

Commençons par démontrer que  $H(u) = \frac{1}{2}H^R(\frac{1}{2}u)$  où  $H^R$  est le polynôme utilisé dans la construction de  $\Gamma^R$ , la courbe de bord avant raffinement. Soient  $a_0^R, a_1^R, a_2^R$  et  $h_0^R, h_1^R, h_2^R$  les points de contrôle de  $\mu^R \cdot \nu^R$  et  $H^R$  dans la base de Bézier. On veut démontrer que

$$\begin{cases} h_0 = \frac{1}{2}h_0^R \\ h_1 = \frac{h_0^R + h_1^R}{4} \\ h_2 = \frac{h_0^R + 2h_1^R + h_2^R}{4 \cdot 2} \end{cases} \quad (2.4.11)$$

On a vu que  $\mu(u) = \mu^R(\frac{1}{2}u)$  et  $\nu(u) = \nu^R(\frac{1}{2}u)$  donc

$$\begin{cases} a_0 = a_0^R \\ a_1 = \frac{a_0^R + a_1^R}{2} \\ a_2 = \frac{a_0^R + 2a_1^R + a_2^R}{4} \end{cases} \quad (2.4.12)$$

et par construction de l'interpolant (équation (2.3.22)) :

$$\begin{cases} h_0 = \frac{5}{a_0}(b_1 - b_0) \\ h_1 = \frac{10}{a_0}(b_2 - b_1) - \frac{a_1}{a_0}h_0 \\ \alpha = \frac{1}{30}(6a_2 + 3a_1 + a_0) \\ \beta = \frac{1}{30}(3a_2h_1 + 4a_1h_1 + a_2h_0) + b_2 \\ h_2 = \frac{\beta^d - \beta}{\alpha + \alpha^d} \end{cases} \quad (2.4.13)$$

On a donc

$$\begin{aligned} h_0 &= \frac{5}{a_0}(b_1 - b_0) \text{ tiré de (2.4.13)} \\ &= \frac{5}{a_0^R} \frac{B_1 - B_0}{2} \text{ équations (2.4.1)} \\ &= \frac{1}{2}h_0^R \text{ grâce à (2.4.13) au niveau précédent} \end{aligned}$$

de même en partant de l'autre extrémité

$$\begin{aligned} h_0^d &= \frac{5}{a_0^d}(b_1^d - b_0^d) \\ &= \frac{5}{a_2^R} \frac{B_4 - B_5}{2} \\ &= -\frac{1}{2}h_2^R \end{aligned}$$

De la même façon

$$\begin{aligned} h_1 &= \frac{10}{a_0}(b_2 - b_1) - \frac{a_1}{a_0}h_0 \text{ tiré de (2.4.13)} \\ &= \frac{10}{a_0^R} \left( \frac{B_2 + 2B_1 + B_0}{4} - \frac{B_0 + B_1}{2} \right) - \frac{a_0^R + a_1^R}{2a_0^R} \frac{1}{2}h_0^R \text{ équations (2.4.1)} \\ &= \frac{1}{4}h_1^R - \frac{10}{4a_0^R}(B_1 - B_0) - \frac{1}{4}h_0^R \text{ grâce à (2.4.13) au niveau précédent} \\ &= \frac{h_0^R + h_1^R}{4} \end{aligned}$$



en partant de l'autre extrémité

$$\begin{aligned}
h_1^d &= \frac{10}{a_0^d}(b_2^d - b_1^d) - \frac{a_1^d}{a_0^d}h_0^d \text{ tiré de (2.4.13)} \\
&= \frac{10}{a_2^R} \left( \frac{B_3 + 2B_4 + B_5}{4} - \frac{B_4 + B_5}{2} \right) + \frac{a_1^R + a_2^R}{2a_2^R} \frac{1}{2} h_2^R \text{ équations (2.4.1)} \\
&= -\frac{h_2^R}{2} + \frac{10}{a_2^R} \frac{B_3 - B_4}{4} + \frac{h_2^R}{4} + \frac{a_1^R}{4a_2^R} h_2^R \text{ grâce à (2.4.13) au niveau précédent} \\
&= -\frac{h_2^R}{4} - \frac{a_1^R h_2^R + a_2^R h_1^R}{4a_2^R} + \frac{a_1^R}{4a_2^R} h_2^R \\
&= -\frac{h_2^R + h_1^R}{4}
\end{aligned}$$

On s'intéresse maintenant à  $\alpha$  et  $\beta$  de la formule (2.4.13) :

$$\begin{aligned}
\alpha &= \frac{1}{30}(6a_2 + 3a_1 + a_0) \text{ tiré de (2.4.13)} \\
&= \frac{1}{30} \left( 6 \frac{a_0^R + 2a_1^R + a_2^R}{4} + 3 \frac{a_0^R + a_1^R}{2} + a_0^R \right) \text{ vu (2.4.12)} \\
&= \frac{1}{30} \frac{8a_0^R + 9a_1^R + 3a_2^R}{2} \\
&= \frac{\alpha^R}{4} + \frac{1}{30} \frac{15a_0^R + 15a_1^R}{4} \text{ vu (2.4.13) au niveau précédent}
\end{aligned}$$

en partant de l'autre extrémité

$$\begin{aligned}
\alpha^d &= \frac{1}{30}(6a_2^d + 3a_1^d + a_0^d) \\
&= \frac{1}{30} \left( 6 \frac{a_0^R + 2a_1^R + a_2^R}{4} + 3 \frac{a_2^R + a_1^R}{2} + a_2^R \right) \\
&= \frac{1}{30} \frac{8a_2^R + 9a_1^R + 3a_0^R}{2}
\end{aligned}$$

donc

$$\begin{aligned}
\alpha + \alpha^d &= \frac{1}{30} \frac{8a_0^R + 9a_1^R + 3a_2^R}{2} + \frac{1}{30} \frac{8a_2^R + 9a_1^R + 3a_0^R}{2} \\
&= \frac{1}{30} \frac{11a_0^R + 18a_1^R + 11a_2^R}{2}
\end{aligned}$$

De même

$$\begin{aligned}
\beta &= \frac{1}{30}(3a_2h_1 + 4a_1h_1 + a_2h_0) + b_2 \text{ tiré de (2.4.13)} \\
&= \frac{1}{30}\left(3\frac{a_0^R + 2a_1^R + a_2^R}{4}\frac{h_0^R + h_1^R}{4} + 4\frac{a_0^R + a_1^R}{2}\frac{h_0^R + h_1^R}{4}\right. \\
&\quad \left.+ \frac{a_0^R + 2a_1^R + a_2^R}{4}\frac{1}{2}h_0^R\right) + \frac{B_2 + 2B_1 + B_0}{4} \text{ d'après (2.4.13)} \\
&= \frac{h_1^R}{30}\frac{11a_0^R + 14a_1^R + 3a_2^R}{4 \cdot 4} + \frac{h_0^R}{30}\frac{13a_0^R + 18a_1^R + 5a_2^R}{4 \cdot 4} + \frac{B_2 + 2B_1 + B_0}{4} \\
&= \frac{\beta^R}{32} + \frac{h_1^R}{30}\frac{22a_0^R + 23a_1^R + 3a_2^R}{32} + \frac{h_0^R}{30}\frac{26a_0^R + 36a_1^R + 9a_2^R}{32} \\
&\quad + \frac{7B_2 + 16B_1 + 8B_0}{32} \text{ vu (2.4.13) au niveau précédent}
\end{aligned}$$

en partant de l'autre extrémité

$$\begin{aligned}
\beta^d &= \frac{1}{30}(3a_2^d h_1^d + 4a_1^d h_1^d + a_2^d h_0^d) + b_2^d \\
&= \frac{1}{30}\left(-3\frac{a_0^R + 2a_1^R + a_2^R}{4}\frac{h_2^R + h_1^R}{4} - 4\frac{a_1^R + a_2^R}{2}\frac{h_2^R + h_1^R}{4}\right. \\
&\quad \left.- \frac{a_0^R + 2a_1^R + a_2^R}{4}\frac{1}{2}h_2^R\right) + \frac{B_3 + 2B_4 + B_5}{4} \\
&= -\frac{h_1^R}{30}\frac{3a_0^R + 14a_1^R + 11a_2^R}{4 \cdot 4} - \frac{h_2^R}{30}\frac{5a_0^R + 18a_1^R + 13a_2^R}{4 \cdot 4} + \frac{B_3 + 2B_4 + B_5}{4}
\end{aligned}$$

donc

$$\begin{aligned}
\beta^d - \beta &= -\frac{h_1^R}{30}\frac{14a_0^R + 28a_1^R + 14a_2^R}{4 \cdot 4} - \frac{h_2^R}{30}\frac{5a_0^R + 18a_1^R + 13a_2^R}{4 \cdot 4} - \frac{h_0^R}{30}\frac{13a_0^R + 18a_1^R + 5a_2^R}{4 \cdot 4} \\
&\quad + \frac{B_3 + 2B_4 + B_5}{4} - \frac{B_2 + 2B_1 + B_0}{4} \\
&= -\frac{h_1^R}{30}\frac{14a_0^R + 28a_1^R + 14a_2^R}{4 \cdot 4} - \frac{h_2^R}{30}\frac{5a_0^R + 18a_1^R + 13a_2^R}{4 \cdot 4} - \frac{h_0^R}{30}\frac{13a_0^R + 18a_1^R + 5a_2^R}{4 \cdot 4} \\
&\quad + \frac{(B_5 - B_4) + 3(B_4 - B_3) + 4(B_3 - B_2) + 3(B_2 - B_1) + (B_1 - B_0)}{4} \\
&= -\frac{h_1^R}{30}\frac{14a_0^R + 28a_1^R + 14a_2^R}{4 \cdot 4} - \frac{h_2^R}{30}\frac{5a_0^R + 18a_1^R + 13a_2^R}{4 \cdot 4} - \frac{h_0^R}{30}\frac{13a_0^R + 18a_1^R + 5a_2^R}{4 \cdot 4} \\
&\quad + \frac{(a_2^R h_2^R) + \frac{3}{2}(a_1^R h_2^R + a_2^R h_1^R) + \frac{4}{6}(a_0^R h_2^R + 4a_1^R h_1^R + a_2^R h_0^R) + \frac{3}{2}(a_0^R h_1^R + a_1^R h_0^R) + (a_0^R h_0^R)}{5 \cdot 4} \\
&= \frac{h_1^R}{30}\frac{11a_0^R + 18a_1^R + 11a_2^R}{4 \cdot 2} + \frac{h_2^R}{30}\frac{11a_0^R + 18a_1^R + 11a_2^R}{4 \cdot 4} + \frac{h_0^R}{30}\frac{12a_0^R + 18a_1^R + 11a_2^R}{4 \cdot 2} \\
&= \frac{h_0^R + 2h_1^R + h_2^R}{30}\frac{11a_0^R + 18a_1^R + 11a_2^R}{4 \cdot 4}
\end{aligned}$$

On a donc finalement

$$\begin{aligned} h_2 &= \frac{\beta^d - \beta}{\alpha + \alpha^d} \\ &= \frac{h_0^R + 2h_1^R + h_2^R}{4 \cdot 2} \end{aligned}$$

On vient de démontrer les équations (2.4.11) c'est-à-dire que  $H(u) = \frac{1}{2}H^R(\frac{1}{2}u)$ .  
A l'aide de cette fonction, on va déterminer les points de contrôle restants.

Il reste à démontrer que  $b_3$ ,  $b_4$  et  $b_5$  calculés par l'interpolant sont les mêmes que les points de contrôle obtenus par subdivision (i.e. que  $b_3 = \frac{B_0+3B_1+3B_2+B_3}{8}$ ,  $b_4 = \frac{B_0+4B_1+6B_2+4B_3+B_4}{16}$  et  $b_5 = \frac{B_0+5B_1+10B_2+10B_3+5B_4+B_5}{32}$ ).

Par construction de l'interpolant (voir section 2.3 et l'équation (2.3.23)) :

$$\begin{aligned} b_3 &= b_2 + \frac{a_0 h_2 + 4a_1 h_1 + a_2 h_0}{30} \text{ vu (2.3.23)} \\ &= \frac{B_0 + 2B_1 + B_2}{4} + a_0^R \frac{h_2^R + 2h_1^R + h_0^R}{8 \cdot 30} + 4 \frac{a_1^R + a_0^R}{30 \cdot 2} \frac{h_1^R + h_0^R}{4} \\ &\quad + \frac{a_0^R + 2a_1^R + a_2^R}{4 \cdot 30} \frac{1}{2} h_0^R \text{ d'après (2.4.1) et (2.4.11)} \\ &= \frac{B_3}{8} + \frac{2B_0 + 4B_1 + B_2}{8} + a_0^R \frac{2h_1^R + h_0^R}{8 \cdot 30} + 4 \frac{a_1^R h_0^R}{30 \cdot 8} + 4a_0^R \frac{h_1^R + h_0^R}{30 \cdot 8} + \frac{a_0^R + 2a_1^R}{8 \cdot 30} h_0^R \\ &= \frac{B_3}{8} + \frac{2B_0 + 4B_1 + B_2}{8} + h_0^R \frac{6a_0^R + 6a_1^R}{8 \cdot 30} + h_1^R \frac{6a_0^R}{30 \cdot 8} \\ &= \frac{B_3}{8} + \frac{2B_0 + 4B_1 + B_2}{8} + \frac{6 \cdot 10}{8 \cdot 30} (B_2 - B_1) + \frac{5 \cdot 6}{8 \cdot 30} (B_1 - B_0) \\ &= \frac{B_3 + 3B_2 + 3B_1 + B_0}{8} \end{aligned}$$

De la même façon

$$\begin{aligned} b_4 &= b_3 + \frac{a_1 h_2 + a_2 h_1}{10} \text{ vu l'interpolant section 2.3 et l'équation (2.3.23)} \\ &= \frac{B_3 + 3B_2 + 3B_1 + B_0}{8} + \frac{a_0^R + a_1^R}{2 \cdot 10} \frac{h_0^R + 2h_1^R + h_2^R}{4 \cdot 2} \\ &\quad + \frac{a_0^R + 2a_1^R + a_2^R}{4 \cdot 10} \frac{h_0^R + h_1^R}{4} \text{ d'après (2.4.1) et (2.4.11)} \\ &= \frac{B_3 + 3B_2 + 3B_1 + B_0}{8} + \frac{2 \cdot 5(B_4 - B_3)}{16 \cdot 10} + \frac{5 \cdot 6(B_3 - B_2)}{16 \cdot 10} \\ &\quad + 3 \frac{2 \cdot 5(B_2 - B_1)}{16 \cdot 10} + 2 \frac{5(B_1 - B_0)}{16 \cdot 10} \\ &= \frac{B_0 + 4B_1 + 6B_2 + 4B_3 + B_4}{16} \end{aligned}$$

et

$$\begin{aligned}
b_5 &= b_4 + \frac{a_2 h_2}{5} \text{ vu l'interpolant section 2.3 et l'équation (2.3.23)} \\
&= \frac{B_0 + 4B_1 + 6B_2 + 4B_3 + B_4}{16} + \frac{a_0^R + 2a_1^R + a_2^R}{4 \cdot 5} \frac{h_2^R + 2h_1^R + h_0^R}{8} \text{ équation (2.4.11)} \\
&= \frac{B_0 + 4B_1 + 6B_2 + 4B_3 + B_4}{16} + \frac{4 \cdot 5(B_4 - B_3)}{4 \cdot 5 \cdot 8} + \frac{6 \cdot 5(B_3 - B_2)}{4 \cdot 5 \cdot 8} \\
&\quad + \frac{4 \cdot 5(B_2 - B_1)}{4 \cdot 5 \cdot 8} + \frac{5(B_1 - B_0)}{4 \cdot 5 \cdot 8} \\
&= \frac{B_0 + 5B_1 + 10B_2 + 10B_3 + 5B_4 + B_5}{32}
\end{aligned}$$

Le calcul est identique pour la seconde moitié de la courbe. On a donc démontré que la courbe calculée par l'interpolant est identique à celle obtenue par subdivision.

### Plan tangent

Nous venons de voir que la courbe de bord reste inchangée lors de sa reconstruction par l'interpolant après subdivision. Néanmoins cela ne suffit pas. En effet, le plan tangent le long de cette courbe ne doit pas changer non plus pour conserver la continuité  $G^1$  globale. Ce plan tangent est défini en section 2.3 à l'aide des équations (2.3.26) :

$$\begin{cases} 2\nu \frac{\partial S}{\partial u_{i+1}} = \varphi \frac{\partial S}{\partial u_i} + V \\ 2\mu \frac{\partial S}{\partial u_{i-1}} = \varphi \frac{\partial S}{\partial u_i} - V \end{cases} \quad (2.4.14)$$

le long des courbes de bord en posant  $V = \mu\nu W$  avec  $W$  polynôme modifié par rapport à [Hahmann et Bonneau, 2003] pour être de degré 3. Ces conditions entraînent que  $W$  exprimé dans la base de Bézier par ses coefficients  $w_0, w_1, w_1^d$  et  $w_0^d$  vérifie l'équation (2.3.27) :

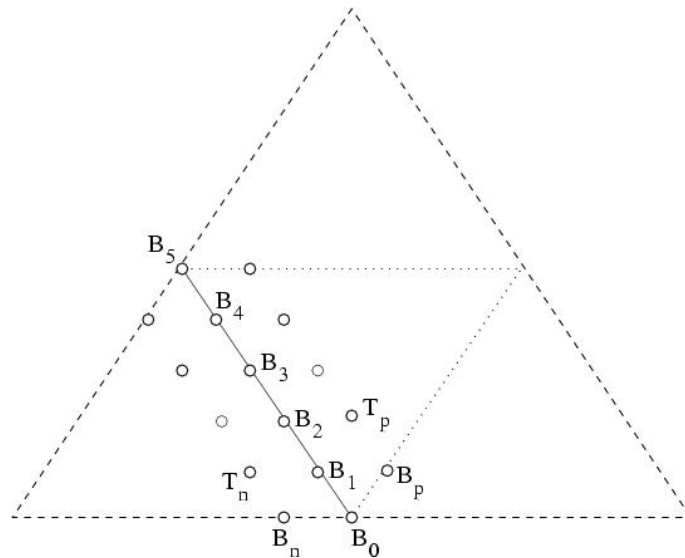
$$\begin{aligned}
w_0 &= \frac{2 \cdot 5}{\nu_0} (b_n - b_0) - \varphi_0 h_0 \text{ vu (2.3.27)} \\
&= \frac{2 \cdot 5}{\nu_0^R} \frac{B_n - B_0}{2} - \frac{1}{2} \varphi_0^R h_0^R \text{ d'après (2.4.1)} \\
&= \frac{1}{2} W_0^R
\end{aligned}$$

et

$$\begin{aligned}
w_1 &= \frac{4 \cdot 5}{\nu_0} (t_n - b_1) - \frac{1}{\nu_0} (\varphi_0 \nu_0 h_1 + \frac{1}{2} (\varphi_0 \nu_1 + \varphi_1 \nu_0) h_0 + \frac{1}{2} (\nu_1 + \nu_0) W_0) \text{ vu (2.3.28)} \\
&= \frac{4 \cdot 5}{\nu_0^R} \frac{T_n - B_1}{4} - \frac{4 \cdot 5}{\nu_0^R} \frac{B_n - B_0}{4} - \frac{1}{\nu_0^R} (\varphi_0^R \nu_0^R \frac{h_1^R + h_0^R}{4} + \frac{1}{2} (\varphi_0^R \frac{\nu_0^R + \nu_1^R}{2} + \frac{\varphi_0^R + \varphi_1^R}{2} \nu_0^R) \frac{1}{2} h_0 \\
&\quad + \frac{1}{2} (\frac{\nu_0^R + \nu_1^R}{2} + \nu_0^R) \frac{W_0^R}{2}) \text{ d'après (2.4.1)} \\
&= \frac{W_1^R}{4} + \frac{4 \cdot 5}{\nu_0^R} \frac{B_n - B_0}{4} - \frac{1}{\nu_0^R} (\frac{1}{2} \varphi_0^R \nu_0^R h_0^R + \frac{1}{2} \nu_0^R \frac{W_0^R}{2}) \text{ vu (2.3.28)} \\
&= \frac{W_0^R + W_1^R}{4}
\end{aligned}$$

Ce qui correspond aux fonctions obtenues par subdivision. De même, en  $v$  le sommet ajouté, ces équations sont encore vérifiées puisqu'elles découlent de la continuité  $G^1$ . On a donc montré que la surface reconstruite par l'interpolant après subdivision partage une même courbe de bord et un même plan tangent le long de celle-ci que la surface obtenue par une simple subdivision. Au voisinage des courbes de bord, le raffinement n'entraîne pas de changement dans la surface. Il faut maintenant étudier le reste de la surface, à commencer par les courbes frontières intérieures qui deviennent des courbes de bord.

### 2.4.2 Erreur le long des courbes intérieures



**Fig. 2.4.4** – Notations utilisées pour l'intérieur.

A l'intérieur du macro-patch avant subdivision, les patches de Bézier se raccorde

de manière  $C^1$ , comme expliqué en section 2.3. Ces courbes intérieures deviennent des courbes frontières après raffinement. Un problème se pose lors de leur recalcul : les fonction scalaires  $\varphi$ ,  $\mu$  et  $\nu$  se simplifiaient du fait de la continuité  $C^1$ , mais apparaissent après raffinement. On les construit donc a posteriori, mais rien ne force la dérivée de ces courbes à avoir ces fonctions en facteur comme le demande l'équation (2.3.21) de la section 2.3. On se doute donc que l'invariance ne sera pas possible. On va néanmoins trouver les points de contrôle invariants, et on va tenter de majorer l'erreur pour les autres.

### Dérivées Secondes

Avec le même style de notation que précédemment mais le long de cette nouvelle courbe frontière, on a :

$$\begin{aligned} B_2 &= B_1 + T_p - B_1 + T_n - B_1 && \text{règle du parallélogramme} \\ &= T_p + T_n - B_1 \end{aligned}$$

et

$$b_2 = \frac{5}{4\varphi_0}A - \frac{\varphi_1}{4\varphi_0}(b_1 - b_0) + b_1 \text{ d'après (2.3.19)}$$

avec  $A$  valant :

$$A = \frac{4}{5}\mu_0(t_p - b_1) + \frac{1}{5}\mu_1(b_p - b_0) + \frac{4}{5}\nu_0(t_n - b_1) + \frac{1}{5}\nu_1(b_n - b_0) \text{ vu (2.3.20)}$$

On va montrer que  $b_2$  est identique à celui obtenu par subdivision i.e.  $b_2 = \frac{B_0 + 2B_1 + B_2}{4}$ . Du fait de la continuité  $C^1$ , on a par construction (voir équation (2.3.12)) :

$$\varphi = \mu = \nu = \frac{1}{2} \quad \text{le long de la courbe frontière.} \quad (2.4.15)$$

Donc

$$b_2 = \frac{5}{4\varphi_0}A - \frac{\varphi_1}{4\varphi_0}(b_1 - b_0) + b_1 \text{ en considérant (2.3.19) et (2.4.15)}$$

avec  $A$  valant :

$$A = \frac{4}{5}\mu_0(t_p - b_1) + \frac{1}{5}\mu_1(b_p - b_0) + \frac{4}{5}\nu_0(t_n - b_1) + \frac{1}{5}\nu_1(b_n - b_0)$$

devient

$$\begin{aligned}
b_2 &= (t_p - b_1) + \frac{1}{4}(b_p - b_0) + (t_n - b_1) + \frac{1}{4}(b_n - b_0) - \frac{1}{4}(b_1 - b_0) + b_1 \\
&= t_p + t_n - b_1 + \frac{1}{4}(b_p + b_n - b_1 - b_0) \\
&= t_p + t_n - b_1 \\
&= \frac{B_0 + 2B_1 + B_2}{4}
\end{aligned}$$

Le point  $b_2$  est donc correct car identique à celui obtenu par subdivision.

### Courbes de bord

Les points de contrôle suivants de la nouvelle courbe de bord dépendent de la fonction  $H$  définie en section 2.3 par l'équation (2.3.21) comme :

$$\Gamma'_1(u) = \mu(u)\nu(u)H(u) = \frac{1}{4}H(u)$$

Comme la fonction  $H$  est de degré deux, la courbe de bord  $\Gamma_1(t)$  sera de degré trois et donc ne pourra pas reproduire exactement celle avant subdivision qui était de degré cinq à cause des points libres intérieurs. Nous allons maintenant essayer de borner l'erreur introduite par un raffinement.

Dans le cas  $\varphi = \mu = \nu = \frac{1}{2}$ , la fonction  $H$  devient (exprimée dans une base de Bézier) :

$$\left\{ \begin{array}{l}
h_0 = 10(b_1 - b_0) \text{ vu (2.3.22)} \\
\quad = 5(B_1 - B_0) \text{ d'après (2.4.1)} \\
h_1 = 20(b_2 - b_1) - h_0 \text{ vu (2.3.22)} \\
\quad = 20b_2 - 30b_1 + 10b_0 \\
\quad = 5(B_2 - B_1) \\
\alpha = \frac{1}{6} \text{ vu (2.3.22)} \\
\beta = \frac{1}{30}\left(\frac{7}{2}h_1 + \frac{1}{2}h_0\right) + b_2 \text{ vu (2.3.22)} \\
\quad = \frac{5B_2 + 1B_0}{6} \\
h_2 = \frac{\beta^d - \beta}{\alpha^d + \alpha} \text{ vu (2.3.22)} \\
\quad = \frac{-B_0 + 0B_1 - 5B_2 + 5B_3 + 0B_4 + B_5}{2}
\end{array} \right. \quad (2.4.16)$$

On peut maintenant calculer tous les points de contrôle de la nouvelle courbe et les comparer avec ceux obtenus par subdivision qui sont  $B_0$ ,  $\frac{B_0+B_1}{2}$ ,  $\frac{B_0+2B_1+B_2}{4}$ ,  $\frac{B_0+3B_1+3B_2+B_3}{8}$ ,  $\frac{B_0+4B_1+6B_2+4B_3+B_4}{16}$  et  $\frac{B_0+5B_1+10B_2+10B_3+5B_4+B_5}{32}$ . Ainsi  $b_3$  :

$$\begin{aligned}
b_3 &= b_2 + \frac{h_2 + 4h_1 + h_0}{60} \text{ vu (2.3.23) section 2.3} \\
&= \frac{B_0 + 2B_1 + B_2}{4} + \frac{-B_0 + 0B_1 - 5B_2 + 5B_3 + 0B_4 + B_5}{120} \\
&\quad + \frac{B_2 - B_1}{3} + \frac{B_1 - B_0}{12} \text{ d'après (2.4.16)} \\
&= \frac{30B_0 + 60B_1 + 30B_2}{120} + \frac{-B_0 + 0B_1 - 5B_2 + 5B_3 + 0B_4 + B_5}{120} \\
&\quad + \frac{40B_2 - 40B_1}{120} + \frac{10B_1 - 10B_0}{120} \\
&= \frac{19B_0 + 30B_1 + 65B_2 + 5B_3 + 0B_4 + 1B_5}{120}
\end{aligned}$$

On a donc l'erreur suivante entre  $b_3$  construit et le point de contrôle correspondant de la courbe subdivisée :

$$\begin{aligned}
b_3 - \frac{B_0 + 3B_1 + 3B_2 + B_3}{8} &= \frac{19B_0 + 30B_1 + 65B_2 + 5B_3 + 0B_4 + 1B_5}{120} \\
&\quad - \frac{15B_0 + 45B_1 + 45B_2 + 15B_3}{120} \\
&= \frac{4B_0 - 15B_1 + 20B_2 - 10B_3 + 0B_4 + 1B_5}{120} \\
&= \frac{4(B_0 - B_1) - 11(B_1 - B_2) + 9(B_2 - B_3)}{120} \\
&\quad + \frac{-1(B_3 - B_4) - (B_4 - B_5)}{120}
\end{aligned}$$

$$\begin{aligned}
\|b_3 - \frac{B_0 + 3B_1 + 3B_2 + B_3}{8}\| &= \left\| \frac{4(B_0 - B_1) - 11(B_1 - B_2) + 9(B_2 - B_3)}{120} \right. \\
&\quad \left. + \frac{-1(B_3 - B_4) - (B_4 - B_5)}{120} \right\| \\
&\leq \frac{4 + 11 + 9 + 1 + 1}{120} \max_{0 \leq i \leq 4} \|B_i - B_{i+1}\| \\
&\leq \frac{26}{120} \max_{0 \leq i \leq 4} \|B_i - B_{i+1}\|
\end{aligned}$$



De la même façon que  $b_3$ , on calcule  $b_4$  :

$$\begin{aligned}
b_4 &= b_3 + \frac{h_2 + h_1}{20} \text{ vu (2.3.23) section 2.3} \\
&= \frac{19B_0 + 30B_1 + 65B_2 + 5B_3 + 0B_4 + 1B_5}{120} \\
&\quad + \frac{-B_0 + 0B_1 - 5B_2 + 5B_3 + 0B_4 - B_5 + 10(B_2 - B_1)}{40} \text{ d'après (2.4.16)} \\
&= \frac{4B_0 + 0B_1 + 20B_2 + 5B_3 + 0B_4 + B_5}{30}
\end{aligned}$$

On a donc l'erreur suivante entre  $b_4$  construit et le point de contrôle correspondant de la courbe subdivisée :

$$\begin{aligned}
b_4 - \frac{B_0 + 4B_1 + 6B_2 + 4B_3 + B_4}{16} &= \frac{4B_0 + 0B_1 + 20B_2 + 5B_3 + 0B_4 + B_5}{30} \\
&\quad - \frac{B_0 + 4B_1 + 6B_2 + 4B_3 + B_4}{16} \\
&= \frac{32B_0 + 0B_1 + 160B_2 + 40B_3 + 0B_4 + 8B_5}{240} \\
&\quad - \frac{15B_0 + 60B_1 + 90B_2 + 60B_3 + 15B_4}{240} \\
&= \frac{17B_0 - 60B_1 + 70B_2 - 20B_3 - 15B_4 + 8B_5}{240} \\
&= \frac{17(B_0 - B_1) - 43(B_1 - B_2) + 27(B_2 - B_3)}{240} \\
&\quad + \frac{7(B_3 - B_4) - 8(B_4 - B_5)}{240} \\
\|b_4 - \frac{B_0 + 4B_1 + 6B_2 + 4B_3 + B_4}{16}\| &= \left\| \frac{17(B_0 - B_1) - 43(B_1 - B_2) + 27(B_2 - B_3)}{240} \right. \\
&\quad \left. + \frac{7(B_3 - B_4) - 8(B_4 - B_5)}{240} \right\| \\
&\leq \frac{17 + 43 + 27 + 7 + 8}{240} \max_{0 \leq i \leq 4} \|B_i - B_{i+1}\| \\
&\leq \frac{102}{240} \max_{0 \leq i \leq 4} \|B_i - B_{i+1}\|
\end{aligned}$$

De la même façon que  $b_4$ , on calcule  $b_5$  :

$$\begin{aligned}
b_5 &= b_4 + \frac{h_2}{10} \text{ vu (2.3.23) section 2.3} \\
&= \frac{4B_0 + 0B_1 + 20B_2 + 5B_3 + 0B_4 + B_5}{30} \\
&\quad + \frac{-B_0 + 0B_1 - 5B_2 + 5B_3 + 0B_4 + B_5}{20} \text{ d'après (2.4.16)} \\
&= \frac{5B_0 + 0B_1 + 25B_2 + 25B_3 + 0B_4 + 5B_5}{60}
\end{aligned}$$

On a donc l'erreur suivante entre  $b_5$  construit et le point de contrôle correspondant de la courbe subdivisée :

$$\begin{aligned}
b_5 - \frac{B_0 + 5B_1 + 10B_2 + 10B_3 + 5B_4 + B_5}{32} &= \frac{B_0 + 5B_2 + 5B_3 + B_5}{12} \\
&\quad - \frac{B_0 + 5B_1 + 10B_2 + 10B_3 + 5B_4 + B_5}{32} \\
&= \frac{5B_0 - 15B_1 + 10B_2 + 10B_3 - 15B_4 + 5B_5}{96} \\
&= \frac{5(B_0 - B_1) - 10(B_1 - B_2) + 10(B_3 - B_4)}{96} \\
&\quad + \frac{-5(B_4 - B_5)}{96} \\
\|b_5 - \frac{B_0 + 5B_1 + 10B_2 + 10B_3 + 5B_4 + B_5}{32}\| &= \left\| \frac{5(B_0 - B_1) - 10(B_1 - B_2) + 10(B_3 - B_4)}{96} \right. \\
&\quad \left. + \frac{-5(B_4 - B_5)}{96} \right\| \\
&\leq \frac{5 + 10 + 10 + 5}{96} \max_{0 \leq i \leq 4} \|B_i - B_{i+1}\| \\
&\leq \frac{40}{96} \max_{0 \leq i \leq 4} \|B_i - B_{i+1}\|
\end{aligned}$$

Si on appelle le maximum entre deux points de contrôle successifs le rayon des points de contrôle :

$$R = \max_{0 \leq i \leq 4} \|B_i - B_{i+1}\| \quad (2.4.17)$$

On a alors :

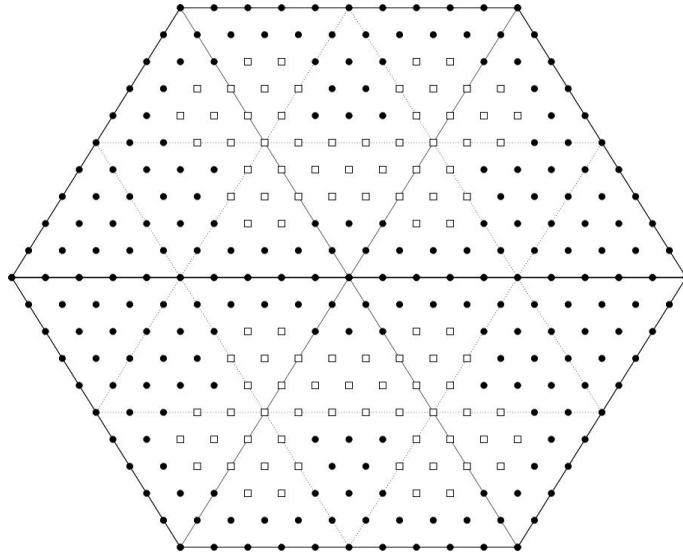
$$\begin{aligned}
\max_{0 \leq j \leq 5} \|b_j - \tilde{b}_j\| &\leq \max\left(\frac{26}{120}, \frac{102}{240}, \frac{40}{96}\right) \max_{0 \leq i \leq 4} \|B_i - B_{i+1}\| \\
&\leq \frac{102}{240} \max_{0 \leq i \leq 4} \|B_i - B_{i+1}\|
\end{aligned}$$

L'erreur maximum sur les points de contrôle est donc de 42.5 % du rayon des points de contrôle.

Bien que cette borne d'erreur soit élevée dans la théorie, plusieurs remarques sont à prendre en considération. Tout d'abord, l'alternance des signes et leur somme qui s'annule permet d'avoir une erreur extrêmement faible dans le cas où les points de contrôle sont répartis presque uniformément. Par ailleurs, les paramètres libres tels les six points libres intérieurs sont obtenus par minimisation d'énergie, ce qui tend justement à placer les points de contrôle  $B_2$  et  $B_3$  uniformément, et les empêchent de présenter de fortes ondulations. Cela diminue radicalement l'erreur atteinte en pratique. De plus, l'erreur ici calculée est une borne sur le polygone de contrôle et

non sur la courbe en elle-même. L'erreur sur la courbe est plus faible car la courbe est obtenue par une fonction combinaison convexe des points de contrôle. L'erreur sur la courbe est donc une fonction linéaire des erreurs, et n'atteint jamais cette valeur maximale. L'erreur donnée ici est une valeur maximale ponctuelle sur les points de contrôle. Enfin, ce calcul ne prend pas en compte le lien existant dans le calcul des différents points de contrôle avant subdivision, les  $B_i$ . Ceux-ci ne sont pas quelconque, mais découlent déjà d'un processus de calcul. En pratique, on observe des erreurs de quelques pourcents, n'atteignant jamais 3 %.

Tout d'abord, la figure 2.4.5 représente l'ensemble des points de contrôle des patches pour remplacer la surface après raffinement, avec ceux que nous avons démontré exacts, et ceux qui introduisent une erreur. On constate que l'erreur est très localisée et ne concerne qu'un ensemble réduit de points de contrôle.



**Fig. 2.4.5** – Erreur introduite par le raffinement. Les cercles représentent les points exacts, les carrés des points introduisant une erreur.

Pour cette raison, l'analyse théorique est délaissée au profit de quelques exemples : celui d'une partie d'un icosaèdre et d'un tore.

La figure 2.4.6 présente les surfaces globales calculées (figures 2.4.6(a) et 2.4.6(b)), les polygones de contrôle de deux patches adjacents (figures 2.4.6(c) et 2.4.6(d)), le résultat d'une subdivision exacte de type Goldman, (figures 2.4.6(f) et 2.4.6(f)) et le résultat du recalcul de l'interpolant (figures 2.4.6(h) et 2.4.6(h)). On constate qu'on ne distingue pas de différence notable à l'œil nu. En effet, l'erreur est respectivement de 1,36 % et 1,29 % ! Cela provient du fait que l'objectif est d'obtenir une surface lisse. En conséquence, on ne peut obtenir les erreurs maximales théoriques qui ne

sont atteintes que pour des polygones fortement ondulés et sans contraintes entre ses points de contrôle, c'est-à-dire des polygones qui ne sont jamais construites. Bien que l'erreur théorique peut atteindre des valeurs trop élevées, en pratique ce comportement n'est jamais observé. On peut tout de même générer des surfaces « bosselées » qui généreront une erreur beaucoup plus élevée. C'est ce que nous faisons en figure 2.4.7. L'erreur devient alors visible à l'œil nu, mais cela dans le cadre d'une surface peu exploitable. Si vraiment ce type de surface doit être obtenu puis raffiné, il est possible de les générer hiérarchiquement, en interprétant chaque bosse comme un détail plus fin. La hiérarchie serait alors constituée d'une surface lisse présentant une courbure uniforme, puis au niveau suivant, chaque bosse est ajoutée car elles sont de la taille du maillage raffiné. En conclusion, bien que certains types de surfaces déclenchent une erreur visible, ils ne sont pas utilisés car pas esthétiques ; et si besoin est, la hiérarchie permet de contourner la difficulté.

## 2.5 Raffinement et édition hiérarchique

Dans la section précédente, la procédure d'un seul raffinement et édition locale est décrite. Cette section traite des différents problèmes qui arrivent lorsque plusieurs raffinement successifs et éditions locales de la surface initiale sont effectués de manière hiérarchique.

### 2.5.1 Aspects topologiques

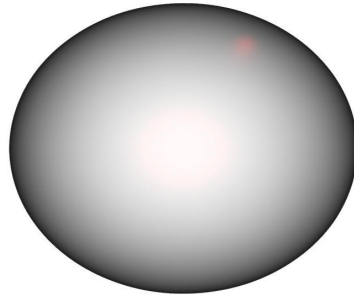
Pendant le processus de design, l'utilisateur fait face à une structure de maillage hiérarchique comprenant des faces, arêtes et sommets à différents niveaux. Soit  $V$  l'ensemble des sommets de la surface à un certain niveau de la hiérarchie et  $V_0$  l'ensemble des sommets du niveau 0, i.e. des sommets du maillage initial.  $V \setminus V_0$  représente les sommets les sommets créés par des subdivision. Chaque sommet étant inséré au milieu d'une arête, il y a équivalence entre les arêtes de la hiérarchie et  $V \setminus V_0$ , les sommets insérés. Chaque arête du maillage est donc représentée par un unique *edge vertex*, sommet d'arête dans la structure de données, voir section 2.2.

Il y a cinq types d'arêtes possibles à chaque instant durant le processus de design (si l'on exclut le cas particulier des bords). La figure 2.5.1 illustre ces cinq types :

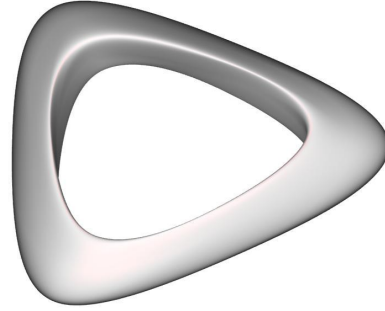
Type 1 : L'arête a deux faces parentes, lesquelles sont subdivisées toutes les deux.

Type 2 : L'arête a deux faces parentes, l'une d'entre elles est subdivisée.

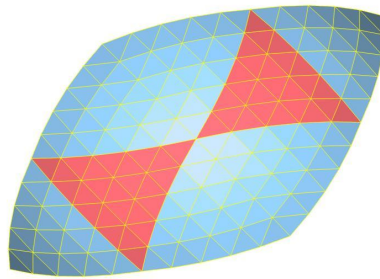
Type 3 : L'arête a deux faces parentes, aucune d'entre elles n'est subdivisée.



(a) Surface définie sur un ico-  
saèdre



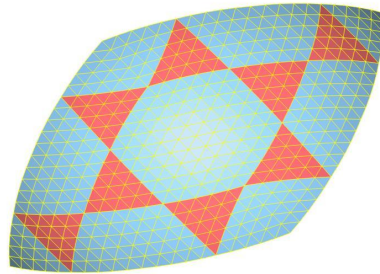
(b) Surface définie sur un tore



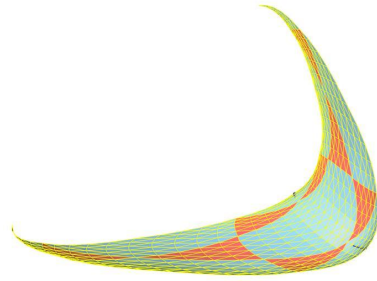
(c) Points de contrôle de l'ico-  
saèdre



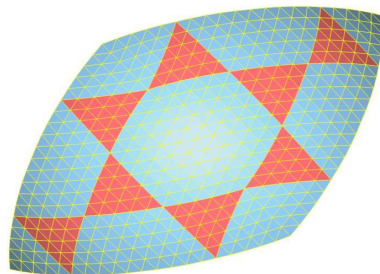
(d) Points de contrôle du tore



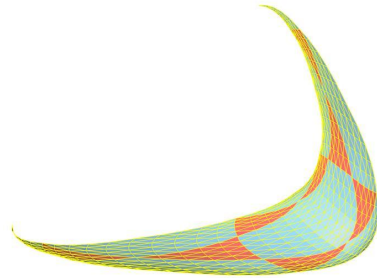
(e) Après une subdivision  
exacte



(f) Après une subdivision  
exacte

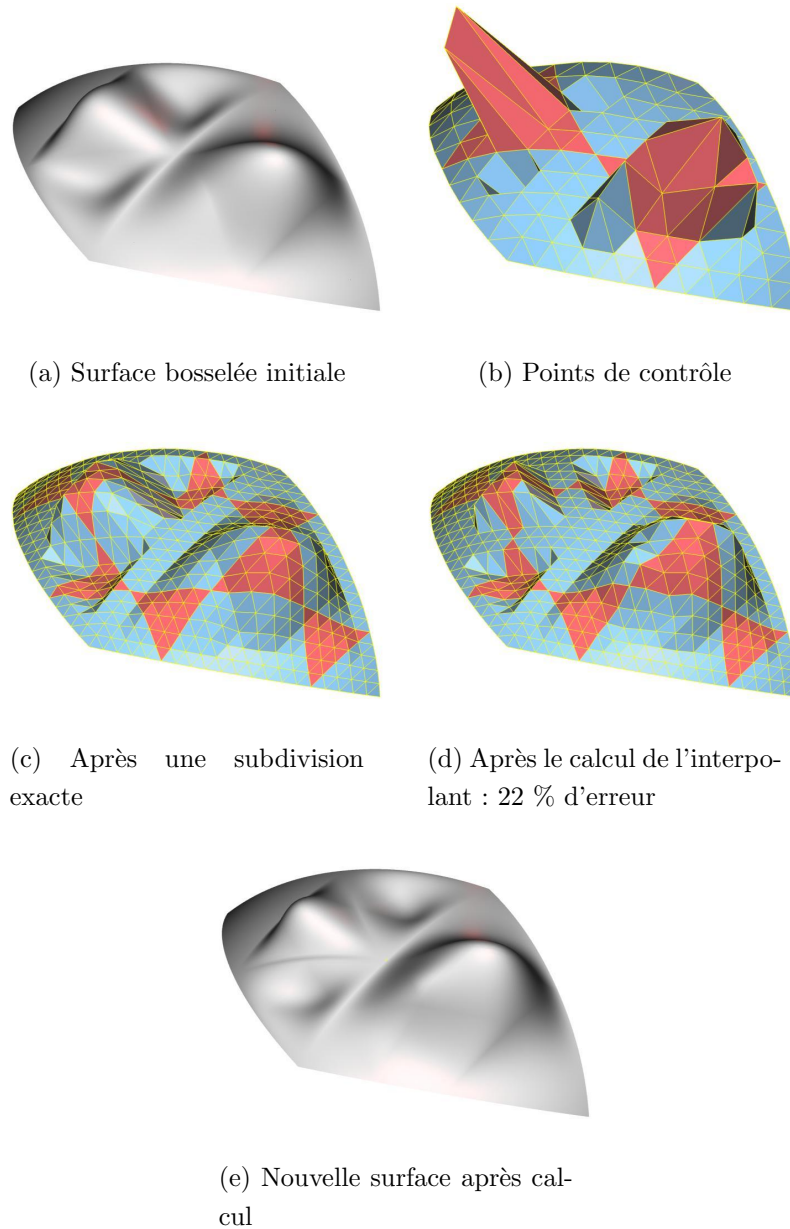


(g) Après le calcul de l'interpo-  
lant : 1,36 % d'erreur

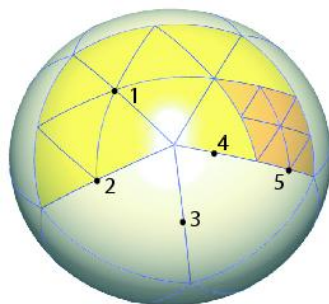


(h) Après le calcul de l'interpo-  
lant : 1,29 % d'erreur

**Fig. 2.4.6** – Observation de l'erreur après une étape de raffinement.



**Fig. 2.4.7** – Observation de l'erreur dans un cas défavorable.

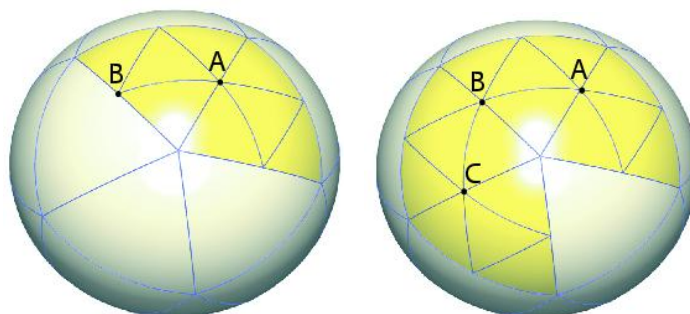


**Fig. 2.5.1** – 5 types d’arêtes se cotoient dans la structure de données hiérarchique

Type 4 : L’arête n’a qu’une face parente, celle-ci n’est pas subdivisée.

Type 5 : L’arête n’a qu’une face parente, celle-ci est subdivisée.

Les arêtes de type 1 correspondent aux sommets éditables. Les arêtes de type 2 et 3 peuvent être raffinées en utilisant la procédure de la section 2.4. Les arêtes de type 4 et 5 ne peuvent pas être raffinées puisque l’une de leurs faces parentes n’existe pas. Avant de les raffiner, leurs faces parentes doivent être créées en subdivisant une arête d’un niveau plus élevé.

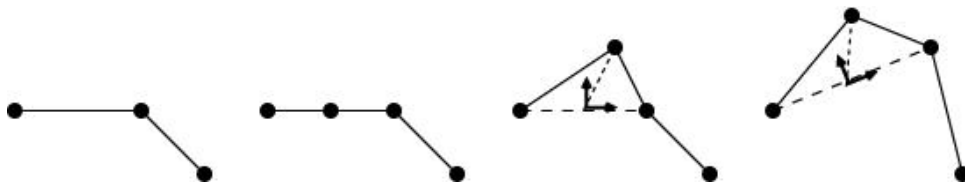


**Fig. 2.5.2** – Après les raffinements successifs des sommets  $A$  et  $C$ , le sommet  $B$  devient éditable.

La condition nécessaire et suffisante pour un sommet d’être éditable est soit d’appartenir au maillage initial  $V_0$ , soit d’avoir deux faces parentes et que ces faces soient subdivisées. Si ce n’est pas le cas, il n’existe pas de maillage triangulaire sous-jacent dans son voisinage. L’interpolant ne peut pas être appliqué. Néanmoins, un sommet non éditable peut le devenir sans que son arête soit explicitement raffinée. Ceci est illustré en figure 2.5.2 : en figure 2.5.2(a), l’arête du sommet  $A$  est raffinée, le sommet  $B$  n’est pas encore éditable. Ensuite en figure 2.5.2(b), l’arête du sommet  $C$  est raffinée, et  $B$  devient éditable.

### 2.5.2 Aspects géométriques

Quand un ou plusieurs paramètres libres d'un sommet éditable sont modifiés à un niveau grossier, tous les patches de la surface affectés par cette modification dans les niveaux plus fins de la hiérarchie doivent être mis à jour. Pour une édition intuitive, les détails encodés dans les niveaux plus fins du modèle hiérarchique doivent suivre le mouvement de la surface au niveau grossier. Un modèle de *reference plus offset* développé par [Forsey et Bartels, 1988] est utilisé pour lier les coordonnées des détails plus fins au niveau précédent de la hiérarchie. Ce lien permet aux détails de suivre dynamiquement l'édition. Chaque paramètre libre (position du sommet, dérivées premières et twists) de la surface autour d'un sommet de niveau  $i$  est défini dans un repère local qui dépend uniquement de la position, du plan tangent et de la normale à la surface au point correspondant au niveau  $i - 1$ . Lors de l'édition d'un sommet, seules les coordonnées locales des paramètres libres de ce sommet changent, pas le repère local. En revanche, tous les repères locaux des niveaux inférieurs dépendant de ce sommet sont mis à jour sans toucher aux coordonnées locales. En conséquence les détails suivent la déformation, tout en conservant leur aspect local. La même idée est utilisée pour encoder les paramètres libres rattachés à l'intérieur des faces (les points libres intérieurs). La figure 2.5.3 illustre ce procédé dans le cas



**Fig. 2.5.3** – Edition Hiérarchique à l'aide d'un repère local.

des courbes polygonales.

## 2.6 Résultats

Différents outils de modélisation sont supportés par notre modèle interpolant hiérarchique. La surface en elle-même propose des degrés de liberté qui peuvent être utilisés pour la modifier intuitivement. Par exemple, les sommets éditables qui sont interpolés par la surface peuvent être sélectionnés sur la surface et déplacés, la surface autour suivant continûment. De plus, à chaque sommet éditable, les directions de tangence des courbes frontières des patches sont libres, à condition de rester dans un même plan. Elles définissent le plan tangent à la surface et les vecteurs normaux



en ces points. En offrant aux designers la possibilité d'interagir directement et interactivement sur ces quantités, différents effets peuvent être obtenus. Modifier la direction normale donne une nouvelle orientation au plan tangent, modifier la longueur du vecteur normal change la tension locale en multipliant toutes les dérivées premières. Un effet de torsion est obtenu en tournant le plan tangent. Certains de ces outils sont illustrés en même temps que le modèleur 3D en figure 3.1.1 section 3.1.

En plus de ces outils, la structure hiérarchique de la surface combinée avec la représentation des parties fines de la surface par des décalages en coordonnées locales permet deux types d'édition :

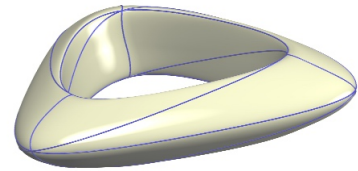
- au niveau le plus fin : l'édition d'un sommet ou paramètre libre du niveau le plus fin modifie la surface localement pour ajouter un détail
- à un niveau plus élevé : l'édition d'un sommet ou paramètre libre d'un niveau plus élevé dans la hiérarchie modifie la surface plus globalement tout en conservant les détails inchangés.

Les étapes successives de l'algorithme sont présentées en figure 2.6.1 avec un maillage de genre 1 (homéomorphe à un tore). Une surface lisse interpolant le maillage de base est montré en figure 2.6.1(a). La figure 2.6.1(b) montre la même surface avec les courbes de bord de ses macro-patches. Cette surface est ensuite raffinée sur deux niveaux en figure 2.6.1(c). Trois sommets les plus fins sont alors édités pour créer la surface de la figure 2.6.1(e). Avec cette technique, la taille des détails est déterminée par le niveau auquel ils sont ajoutés/crétés. Finalement, un sommet du niveau 0 est édité pour illustrer une édition hiérarchique à grande échelle. Les détails créés précédemment suivent harmonieusement le déplacement, voir figure 2.6.1(g).

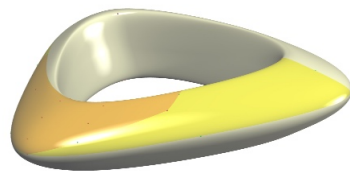
La figure 2.6.2 montre le résultat d'une session de design créant un personnage complexe. A partir d'un icosaèdre (20 faces, 12 sommets) comme maillage initial, la surface interpolante ressemble à une sphère (figure 2.6.2(a)). Avec l'édition d'un sommet, l'utilisateur ajoute un museau (figure 2.6.2(b)). En raffinant le bout du museau et la tête, des oreilles et une gueule sont ajoutées (figure 2.6.2(c)). En raffinant encore plus, les oreilles et la gueule sont améliorées avec plus de détails (2.6.2(d)). Pour finir cette tête de chien, quatre crocs sont ajoutés dans la gueule (figure 2.6.2(i)). Cette gueule peut alors être fermée en éditant un unique sommet suffisamment haut dans la hiérarchie (le bout du museau, figure 2.6.3).



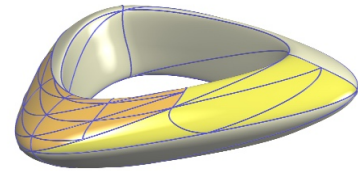
(a) Surface de base (niveau 0)



(b) Patches de la surface de base



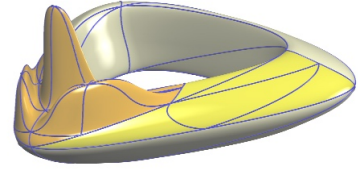
(c) Surface après deux raffinements



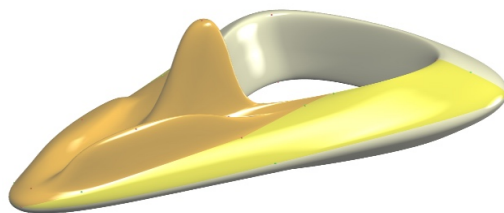
(d) Patches après deux raffinements



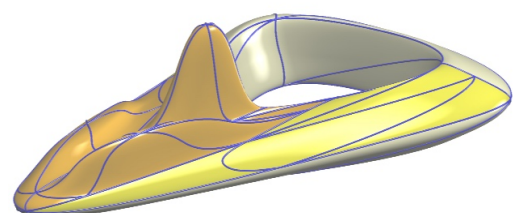
(e) Modifications fines (au niveau 2)



(f) Patches après modifications fines

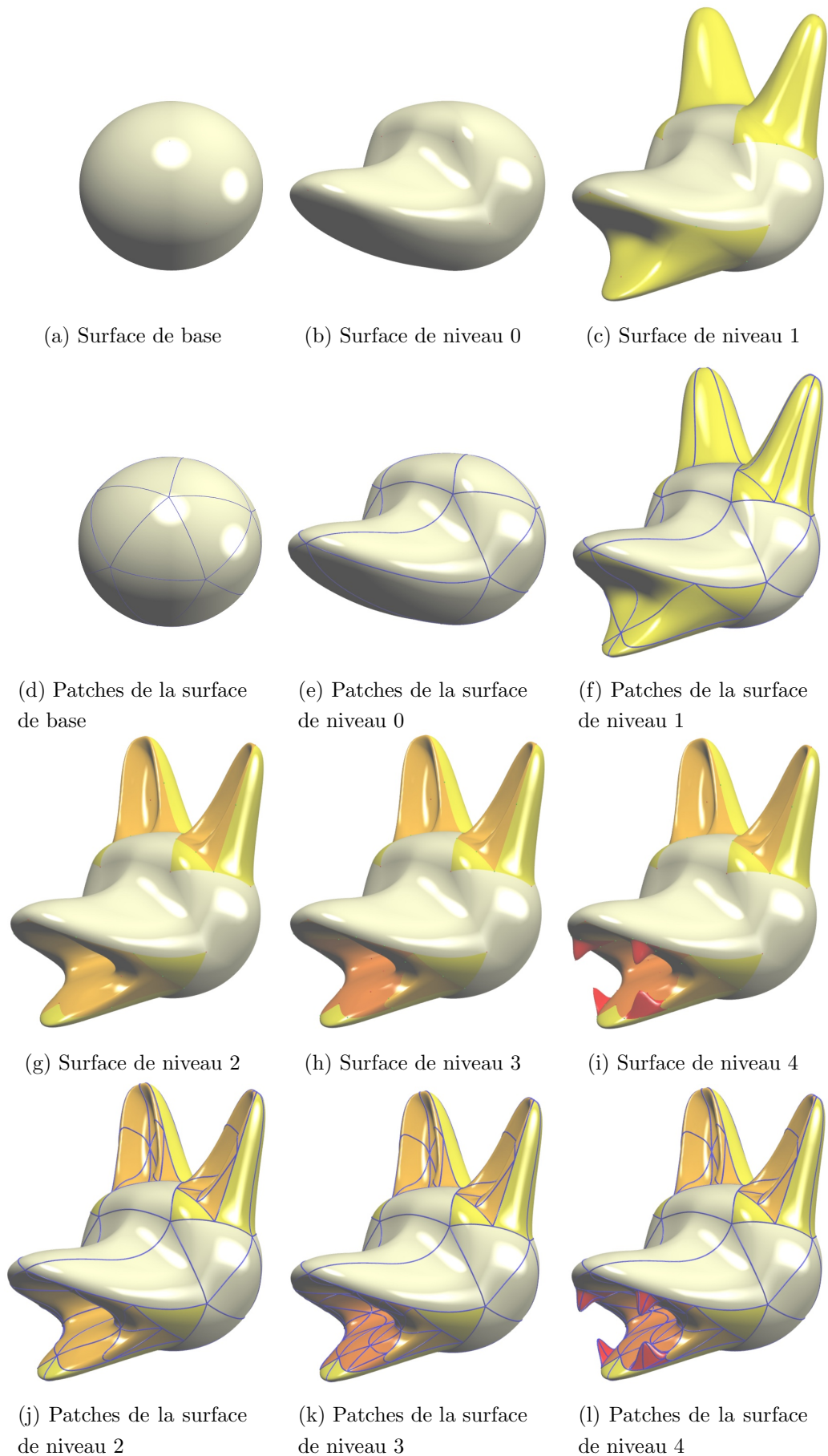


(g) Modification au niveau 0

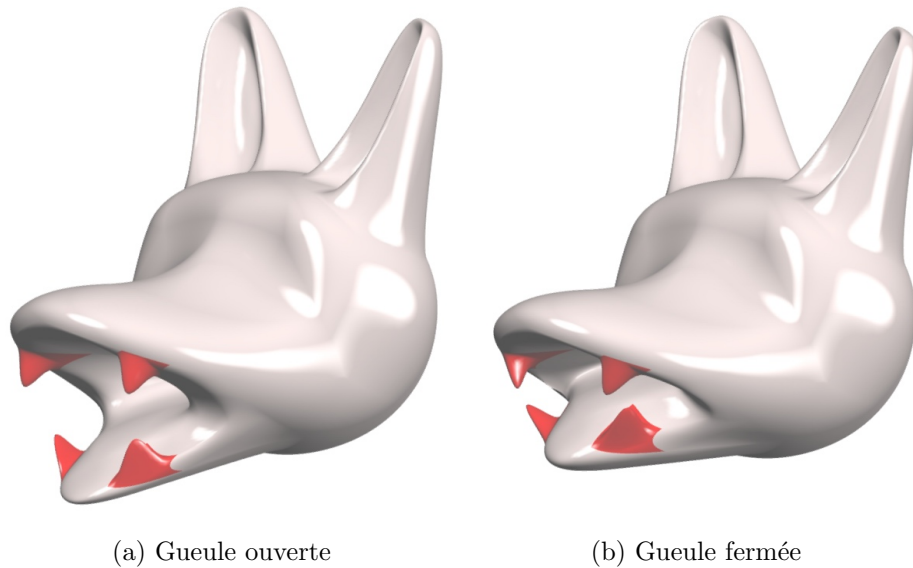


(h) Patches après modification globale

**Fig. 2.6.1** – Surface lisse à partir d'un maillage de genre 1 : raffinements, éditions locales et éditions globales.



**Fig. 2.6.2** – Modèle d'une tête de chien.



**Fig. 2.6.3** – Edition globale : Un seul sommet est modifié.

## Conclusion

Un nouveau modèle hiérarchique de surface a été introduit. Similaire aux H-splines mais traitant les topologies arbitraires, nous avons développé ce schéma interpolant muni de niveaux de détails. Cette méthode fonctionne avec n'importe quel maillage triangulaire représentant une variété de dimension 2 : avec ou sans bord, de genre arbitraire, avec des sommets de degré quelconque. Il présente l'avantage par rapport aux LeSS d'être fondé sur une hiérarchie de surface qui permet une édition progressive continue. Par ailleurs, de nombreux paramètres libres sont offerts à l'utilisateur pour modifier la surface. De plus, aucune propriété n'est imposée au maillage sous-jacent.

Notre modèle simplifie le processus de design grâce à la construction hiérarchique : un maillage initial est tout d'abord créé, représentant une approximation grossière de l'objet et déterminant sa topologie. Une surface lisse initiale est calculée sur ce maillage munie de paramètres libres pour changer sa forme globale. Des détails plus fins sont ensuite ajoutés où besoin est grâce au raffinement local. Ce modèle permet de déformer facilement des objets à grande échelle tout en préservant les détails locaux en manipulant seulement peu de points d'un niveau élevé dans la hiérarchie. Ce modèle peut aussi être utilisé avec un rendu adaptatif : seuls les détails réellement visibles (en fonction du point de vue) sont affichés, ou une fréquence d'affichage minimum garantie : pour un objet trop complexe, certains détails fins ne seraient pas affichés pour ne pas ralentir le rendu.

Ce modèle combine les avantages des surfaces de type produit tensoriel utilisées en CFAO puisqu'il utilise une représentation paramétrique polynomiale, avec les avantages des surfaces de subdivision puisqu'il manipule toute topologie arbitraire et des niveaux de détail.

# Chapitre 3

## Applications

### Sommaire

---

<b>Introduction</b> . . . . .	<b>88</b>
<b>3.1 Modeleur 3D</b> . . . . .	<b>89</b>
<b>3.2 Reconstruction de surface</b> . . . . .	<b>92</b>
3.2.1 Reconstruction de surfaces linéaires . . . . .	95
Application à des données topographiques . . . . .	95
Reconstruction à partir d'un ensemble de points . . . . .	97
3.2.2 Reconstruction de surfaces lisses . . . . .	99
Calcul d'une paramétrisation . . . . .	102
Maillage grossier . . . . .	103
Courbes frontières . . . . .	104
Chemin de longueur minimale . . . . .	105
Lissage . . . . .	106
Paramétrisation . . . . .	107
Echantillonnage . . . . .	109
Amélioration de l'échantillon . . . . .	110
Reconstruction avec un interpolant hiérarchique . . . . .	111
Surface initiale . . . . .	113
Ajout de détails . . . . .	115
Raffinement global . . . . .	115
Raffinement local . . . . .	115
Résultats . . . . .	116
3.2.3 Conclusion . . . . .	117
<b>Conclusion</b> . . . . .	<b>123</b>

---

## Introduction

Après avoir développé dans la partie 2 une nouvelle modélisation de surfaces par interpolation hiérarchique, nous nous proposons dans ce chapitre d'expliciter deux applications différentes de cette méthode. L'objectif premier de ce travail était de fournir aux designers un outil de création intuitive de surface, facile d'utilisation, dans un cadre de modélisation géométrique. Dans un premier temps, un modèleur 3D permettant la création de surfaces par niveau de détail a été développé. Il présente l'avantage de correspondre à la démarche créative artistique puisque son utilisation repose sur le calcul d'une forme globale progressivement affinée pour obtenir l'objet désiré. De plus, cette structure hiérarchique offre aux artistes et designers la possibilité d'éditer un objet de manière intuitive en ne modifiant que très peu de paramètres.

Dans un second temps, une technique de reconstruction d'objet par notre interpolant hiérarchique a été mise au point. Le principe est de modéliser des objets existants à l'aide de notre représentation hiérarchique. Ces objets sont soit des objets réels concrets, soit des objets modélisés par d'autres méthodes. Un objet réel est parcouru par un scanner 3D qui génère des points de  $\mathbb{R}^3$  appartenant à l'objet. Pour connaître l'objet dans tous ses détails, des millions de points sont créés. Un pré-traitement fournit un maillage. Un objet représenté par une autre méthode comme celles vu en chapitre 1 est exporté sous la forme d'un maillage triangulaire. Ceci ne pose aucune difficulté dans la mesure où l'affichage par la carte graphique s'effectue à travers des triangles. Chaque méthode propose donc une fonction d'exportation à travers des triangles. Néanmoins, ces maillages sont beaucoup trop gros pour être facilement transmis, affichés ou édités. La même surface représentée par notre modèle hiérarchique serait au contraire très pratique : la précision de l'affichage adaptée en choisissant le niveau de détail, l'édition aisée grâce à la structure hiérarchique. L'objectif est donc de modéliser l'objet représenté par cet énorme maillage à l'aide de notre modèle pour profiter de tous ses avantages. Cette étape s'appelle la reconstruction de surface. Elle est primordiale car elle permet d'exploiter les avantages de notre modèle sur des objets déjà construits numériquement ou physiquement existants dont des données seraient extraites. Cette procédure permet d'utiliser la modélisation numérique sur des objets réels dans le but de les améliorer ou de les intégrer à une scène virtuelle.

## 3.1 Modeleur 3D

Basé sur l'interpolant hiérarchique qui vient d'être présenté, un modeleur 3D a été implémenté en C++ et OpenGL. L'utilisateur peut charger un maillage, calculer une surface sur ce maillage, modifier cette surface, la raffiner localement, éditer la surface au niveau le plus fin et à n'importe quel niveau de la hiérarchie. Ce programme permet le design et l'édition interactive de surfaces complexes. En fait, les mises à jour (recalcul et affichage) suite à une modification de la surface par un déplacement d'un sommet ou changement de n'importe quel paramètre libre prennent moins de 10 ms sur un ordinateur portable à 1.4 GHz. Cette vitesse ne dépend pas du nombre total de patches (la localité rend inutile de recalculer les patches éloignés), mais du nombre de faces affectées par l'édition et de la profondeur du sous-arbre des faces à parcourir. Cette vitesse de 10 ms est obtenue dans le pire des cas pour une hiérarchie de 5 niveaux. Dans ce cas précis, une surface test a été raffinée 4 fois. Chaque patch du niveau 0 de la hiérarchie est alors remplacé par  $4^4 = 256$  patches plus fins. Le sommet du niveau 0 est alors édité. Comme ce sommet était commun à 5 patches de la surface, le nombre total de patches qui a été recalculé est donc de  $1280 = 5 \times 256$ .

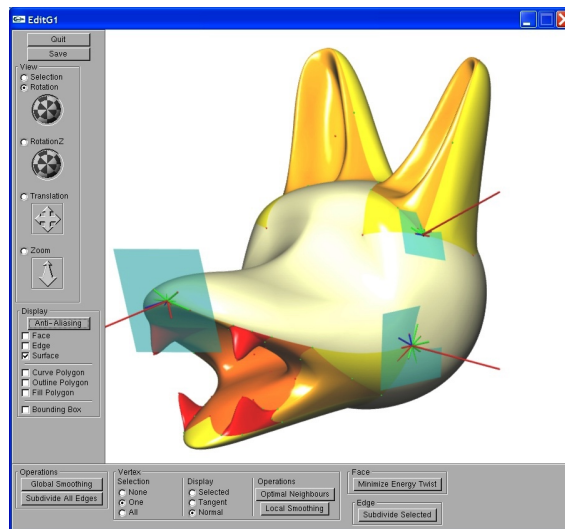


Fig. 3.1.1 – Modeleur 3D

La figure 3.1.1 présente l'interface utilisateur du modeleur développé. Deux boutons dans le coin supérieur gauche permettent de quitter l'application ou de sauver le résultat d'une session de design dans un fichier. Pour cela, un nouveau format de fichier a été développé qui prend en compte l'ensemble des paramètres nécessaires en vue de la construction de la surface. Ce format est détaillé dans l'annexe B. Immédiatement en dessous de ces boutons, un panneau permet de modifier le point



de vue de la fenêtre de rendu (zoom, translation, rotations) et de passer en mode de sélection. Dans ce mode, toutes les parties constituant la surface, le maillage et ses paramètres libres peuvent être sélectionnées en cliquant directement dessus du bouton gauche de la souris. Les opérations ultérieures seront appliquées uniquement aux objets sélectionnés. Un deuxième panneau se trouve sous celui gérant le point de vue. Il offre le choix des composantes à afficher : arêtes et faces du maillages, surface, courbes de bord, polygône des points de contrôle de la surface et ses arêtes, boîte englobante. Tous ces objets graphiques ne sont affichés qu’au voisinage d’éléments sélectionnés. Pour afficher une face du maillage, il faut donc que l’un de ses trois sommets, l’une de ses arêtes ou la face elle-même soit sélectionné. Un bouton permet d’effectuer un anti-crénelage ponctuel pour faire des copies d’écran. Dans le bas de la fenêtre apparaissent les contrôles des principales commandes, en particulier le bouton lançant la subdivision des arêtes sélectionnées et celui optimisant les paramètres libres en vue du lissage de la surface. Lors de l’appui sur le bouton de subdivision, chaque arête sélectionnée est subdivisée et ses faces voisines coupées en quatre faces filles. L’appui sur le bouton d’optimisation permet quant à lui de calculer des points twists optimaux pour une énergie de lissage autour des sommets sélectionnés. Les autres contrôles affichent graphiquement les paramètres libres pour les modifier à la souris. Ainsi, un sommet sélectionné peut-être dans l’état *Selected*, *Tangent* ou *Normal*. L’état *Selected* rend possible son édition, c’est-à-dire son déplacement par un cliquer-déplacer du bouton droit de la souris. L’état *Tangent* affiche les dérivées premières autour du sommet. Celles-ci sont modifiables elles-aussi par le bouton droit de la souris. L’état *Normal* rajoute le plan tangent au sommet en translucide et le vecteur normal, que l’on peut modifier à nouveau du bouton droit de la souris. Les autres paramètres libres sont optimisés par défaut par un critère de lissage mais restent accessibles avec de nombreuses autres fonctions par leur raccourci clavier. L’interface n’en propose pas l’accès pour faciliter le travail du novice. Une deuxième interface orientée expert est à l’étude. Le chargement, soit d’un maillage initial, soit d’une session sauvegardée, s’effectue par la ligne de commande au moment du lancement du logiciel. L’ensemble des couleurs, propriétés des lumières, du matériau, paramètres d’affichage sont lus au lancement dans un fichier suffixé ini.

La figure 3.1.2 regroupe quelques exemples de modifications réalisées à travers ce modeleur. 3.1.2(a) est un maillage dont les faces et les arêtes sont affichées. 3.1.2(b) montre la partie de la surface au voisinage des éléments sélectionnés ainsi que les courbes de bord après quelques étapes de subdivision. La figure 3.1.2(c) illustre le polygône de contrôle surligné. 3.1.2(d) présente les paramètres libres autour d’un sommet : les dérivées premières, twists, normale et plan tangent. L’une des dérivées

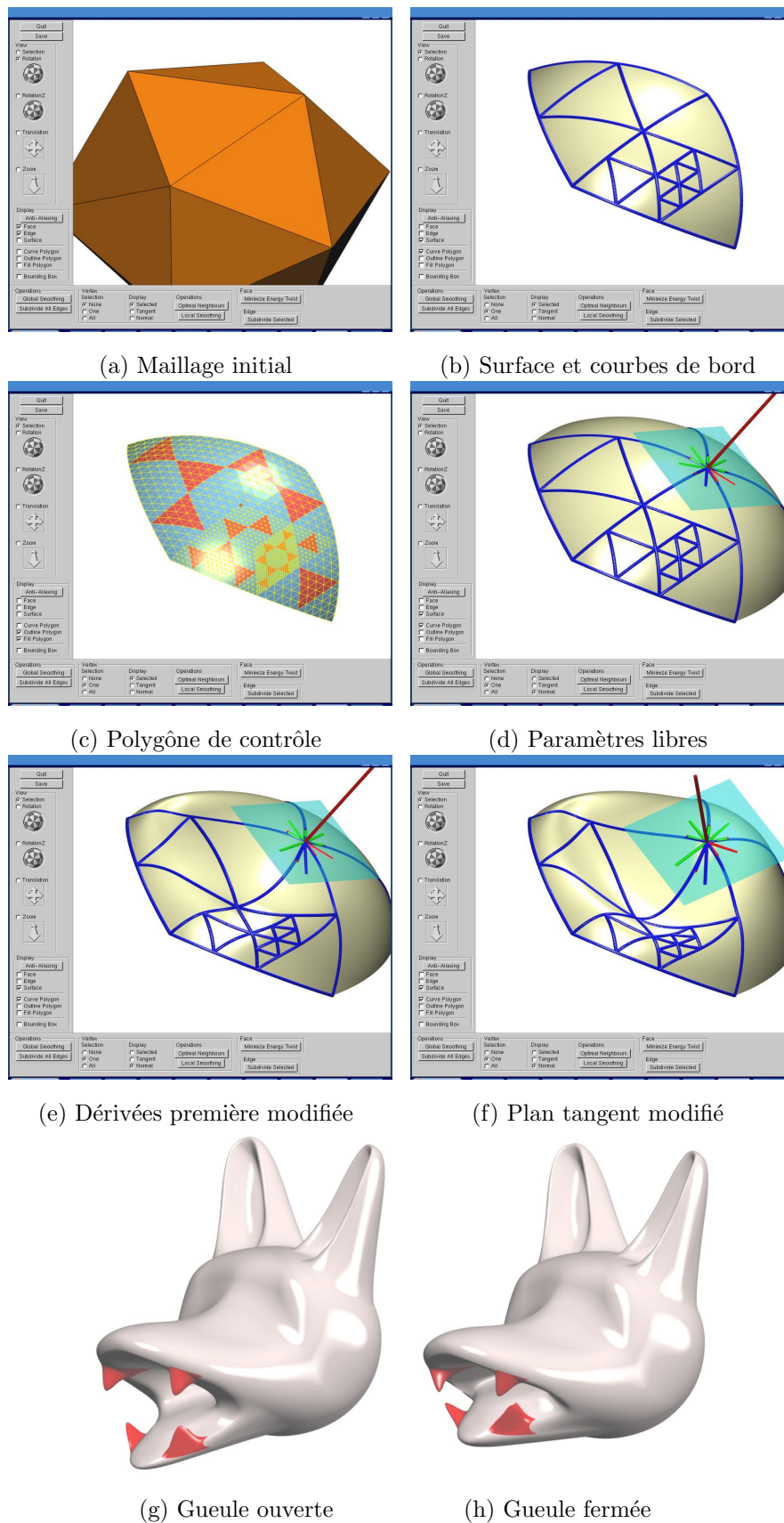


Fig. 3.1.2 – Exemples d'opérations réalisées par notre modeleur.

premières a été éditée dans 3.1.2(e). 3.1.2(f) est obtenue après édition de la normale. L'édition hiérarchique entre les figures 3.1.2(g) et 3.1.2(h) ont été obtenues par l'édition d'un unique sommet situé au bout de la mâchoire inférieure. Notre modelleur permet donc de pleinement exploiter la modélisation interpolante lisse hiérarchique développée dans la partie 2.

## 3.2 Reconstruction de surface

Bien que la modélisation informatique soit toujours utilisée, beaucoup de designers préfèrent toucher de l'argile ou du bois en réalisant des maquettes. Ces maquettes doivent alors être réintégréées dans l'ordinateur. Cela se fait principalement à l'aide de scanners laser 3D. Cependant, ceux-ci génèrent des nuages de points ou des maillages de plus en plus gros. Il est ainsi courant de manipuler des maillages de plusieurs millions de faces, et même dizaines de millions. Ces maillages sont si gros qu'il devient difficile de les stocker, transmettre, afficher et surtout éditer. Un maillage de 10 millions de faces est devenu chose courante. Certains maillages ne tiennent même pas en mémoire centrale. Pour l'affichage, les cartes graphiques affichent couramment 10 millions de faces par seconde. La fréquence minimale perçue comme fluide par l'oeil humain est de 25 Hz, c'est-à-dire 25 images par secondes. Pour changer le point de vue de manière fluide, il ne faut pas dépasser  $10.000.000/25 = 400.000$  faces. Quelle que soit l'utilisation de ce maillage, le problème sera du même type. Par ailleurs, ces maillages présentent les inconvénients communs à toute modélisation par maillage vus en chapitre 1.4, à savoir un nombre fixe de faces qui sont parfois redondantes dans les zones plates, parfois inutiles si l'objet est loin du point de vue, et parfois insuffisantes pour la silhouette en cas de zoom. Il faut donc proposer des manipulations sur ces maillages pour pouvoir les employer. La solution généralement adoptée est l'emploi de différentes techniques de modélisation d'objet appliquées à ces données. Cela s'appelle l'ingénierie inverse, ou la reconstruction. Le principe de la reconstruction est donc de décrire un ensemble de données constitué de points ou un maillage à l'aide d'une des méthodes présentées dans la partie 1 par exemple. Cela représente en soi un vaste domaine de recherche, le cas particulier de chaque méthode devant être étudié à part.

Cependant, le processus reste le même : une correspondance entre un point de la surface modélisée et un point des données est tout d'abord trouvée. Cette étape est primordiale et influence très nettement le résultat. En effet, une erreur est calculée une fois cette liaison établie. Cette erreur représente la précision avec laquelle la sur-

face reproduit les données. Une erreur relative faible signifie que la surface engendrée présente la même forme que les données : c'est l'objectif que ces méthodes cherchent à atteindre. Pour y arriver, les paramètres libres de la modélisation de surfaces sont mis à contribution. Chaque point de la surface intervenant dans le calcul de l'erreur est écrit en fonction de ces paramètres libres. L'erreur en fonction des paramètres libres est alors déduite. Le but étant de réduire l'erreur, une minimisation de la fonction d'erreur par rapport aux paramètres libres est alors effectuée. Les paramètres libres optimaux pour cette erreur et ces données sont ainsi générés.

Dans le cas où les données sont constituées d'un ensemble de points, une triangulation est parfois demandée. C'est le choix que nous avons fait. Il paraît en effet naturel de prendre un maillage triangulaire en entrée car cela constitue les briques de base utilisées par la carte graphique. L'affichage d'une surface s'effectue à travers une évaluation générant des triangles. Toutes les méthodes présentées au chapitre 1 fournissent donc des outils permettant d'obtenir des triangles. La reconstruction d'objets déjà modélisés selon une certaine technique se fait donc directement. Dans le cas de données issues d'un scanner 3D et fournies sous la forme d'un ensemble de points, plusieurs méthodes existent pour les transformer en maillage. Parmi celles-ci, citons [Curless et Levoy, 1996, Hoppe *et al.*, 1993, Turk et Levoy, 1994]. M. Floater propose quant à lui de relier un nombre constant de plus proches voisins puis de paramétrer cette entité, c'est-à-dire en construire une image dans le plan. La triangulation de cette image plane par une triangulation de Delaunay fournit une triangulation spatiale des points. Cette technique présente l'intérêt de pouvoir être localisée, un petit travail de raccord permettant de recoller les triangulations locales. Plusieurs techniques coexistent donc. En conclusion, la contrainte d'avoir en entrée une triangulation n'en est pas vraiment une.

L'analyse multirésolution par ondelettes a été utilisée dans [Eck *et al.*, 1995]. Une surface multirésolution dans une base d'ondelettes est contruite pour reproduire le maillage fin initial. Le principal obstacle à l'utilisation des surfaces de topologie quelconque par ondelettes est la forme particulière des maillages générés. Un maillage initial est subdivisé par une règle de quatre : chaque face génère quatre faces filles. Cette particularité s'appelle la propriété de connexion de subdivision. Seuls des maillages possédant cette propriété se reconstruisent sans effort, et cela en fait peu. Pour les autres, des calculs sont nécessaires pour créer des données de cette forme. [Eck *et al.*, 1995] détaille ces traitements : l'obtention du maillage initial puis de la surface dans la base d'ondelettes en majorant l'erreur relative. Le maillage initial découle d'une procédure d'ensemencement. Une graine est semée, c'est-à-dire un sommet du maillage marqué comme sommet du maillage initial. Ce maillage initial

est appelé grossier car il ne représente que grossièrement la forme de l'objet. Toutes les distances des autres sommets à celui-là sont calculées. Le sommet le plus éloigné est marqué à son tour, et ainsi de suite. Un diagramme de Voronoï permet de séparer le maillage en parties les plus proches des sommets marqués. Ce diagramme indique comment relier les sommets. Ils sont alors connectés par des plus courts chemins et un échantillonnage respectant la propriété de connexion de subdivision est généré. La reconstruction de cet échantillon par une surface multirésolution est alors directe. Les résultats de cette méthode sont intéressants mais l'impossibilité pour l'utilisateur d'intervenir est gênante car il ne peut pas adapter le résultat à l'utilisation future de son objet.

[Lee *et al.*, 1998] optimise et automatise au maximum le processus de reconstruction par des ondelettes en profitant du principe de multirésolution. Globalement, le maillage grossier est obtenu par décimation du maillage fin, c'est-à-dire suppression itérative de sommets. Lors de chaque suppression, le sommet supprimé est exprimé comme une combinaison linéaire de ses voisins les plus proches. Après simplifications, chaque sommet supprimé est connu comme une combinaison linéaire des sommets restants. Ceux-ci forment le maillage grossier. Cela constitue une paramétrisation du maillage fin sur le maillage grossier. Ce travail effectué, il devient facile de calculer une surface multirésolution par évaluation du point à insérer par la paramétrisation.

[Krishnamurthy et Levoy, 1996] ne calcule quant à lui aucune paramétrisation explicite pour reconstruire un objet avec des surfaces lisses, donnant l'exemple des B-splines. Une méthode itérative est employée. L'idée est de créer une B-spline très approximative des données, et de regarder le point appartenant à la surface le plus proche de chaque point des données. La surface est alors optimisée dans le sens du rapprochement de ces points. Et on recommence : de nouvelles correspondances sont générées puis la surface optimisée et ainsi de suite. Cette méthode est numériquement rapide, mais présente comme inconvénient de ne pas gérer toutes les topologies et de n'avoir que peu de fondements théoriques.

Plusieurs carreaux de B-splines sont générés dans [Eck et Hoppe, 1996] pour gérer les cas des topologies quelconques. En fait, la paramétrisation utilisée dans [Eck *et al.*, 1995] est ici mise à profit. Les faces triangulaires du maillage grossier sont alors regroupées deux à deux pour former des carreaux sur lesquels des carreaux de B-splines sont construits puis optimisés pour reproduire les données en prenant en compte des contraintes de continuité.

Les splines hiérarchiques (H-splines) de [Forsey et Bartels, 1988] sont utilisées dans [Forsey et Bartels, 1995, Forsey et Wong, 1998] à l'aide d'une optimisation sur les paramètres libres, c'est-à-dire les points de contrôle de la B-spline, après calcul

d'une paramétrisation.

[Litke *et al.*, 2001] et [Hoppe *et al.*, 1994] manipulent quant à eux des surfaces de subdivision mais utilisent des paramétrisations déjà présentées ([Eck *et al.*, 1995, Lee *et al.*, 1998]). On constate que l'étape primordiale de toute reconstruction est la paramétrisation du maillage fin sur un maillage simple.

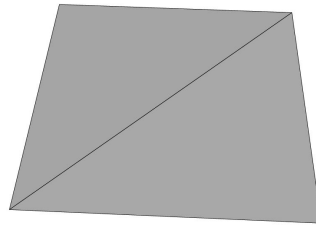
Bien que ces méthodes permettent effectivement la reconstruction 3D, elles souffrent toujours des inconvénients inhérents aux techniques de modélisation sous-jacentes comme vu au chapitre 1. Certaines se cantonnent à des objets homéomorphes à un disque, tore, ou cylindre. Comme cette contrainte est très réductrice du fait des topologies variées rencontrées sur des objets réels, ces méthodes sont étendues pour traiter des objets de topologie quelconque. Notre schéma d'interpolation hiérarchique se présente comme une base mieux adaptée pour reconstruire des surfaces notamment grâce à ses propriétés d'interpolation, de hiérarchie, et ses paramètres libres. Dans cette section, la méthode utilisée pour reconstruire des données par une surface hiérarchique est détaillée. Ce chapitre commence par une première approche de la reconstruction à travers le cadre de surfaces hiérarchiques linéaires par morceaux, c'est-à-dire de maillages qui subissent une subdivision en quatre. Le point inséré en milieu d'arête étant libre, il est déplacé pour ajouter des détails. C'est le seul paramètre libre de ce modèle. Cette approche permet de tester nos algorithmes de construction dans des cas simples ne nécessitant pas de paramétrisation. Par la suite, nous étendrons notre démarche au cadre des surfaces lisses hiérarchiques développées en chapitre 2.

### 3.2.1 Reconstruction de surfaces linéaires

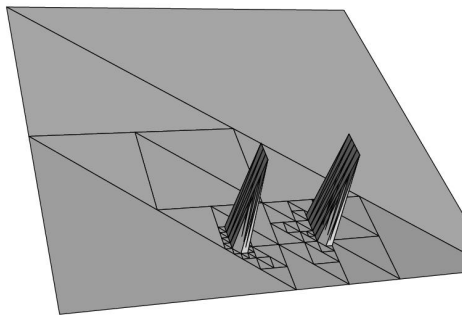
Pour tester le principe de la reconstruction hiérarchique d'objets, le cas particulier des surfaces hiérarchiques linéaires par morceaux avec données régulières sur une grille est abordé dans le cadre de données topographiques et d'un ensemble de points. L'objectif de cette partie est de vérifier le comportement du modèle hiérarchique et de ses raffinements pour représenter un objet. La modélisation de surface est choisie linéaire car elle offre un cadre d'étude simple avant d'aborder le cas complexe des surfaces lisses de continuité  $G^1$ .

#### Application à des données topographiques

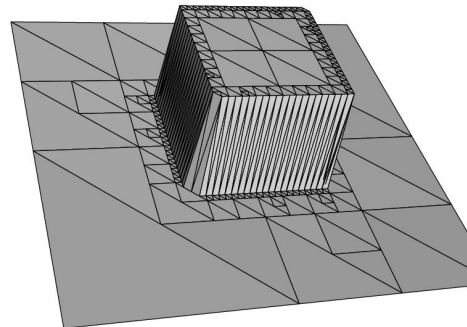
Des données topographiques des Etats-Unis sont disponibles sur un site du gouvernement américain. Nous avons donc exploité ces données pour reconstruire des zones de la carte. Après traitement, un tableau des altitudes est généré, chaque point



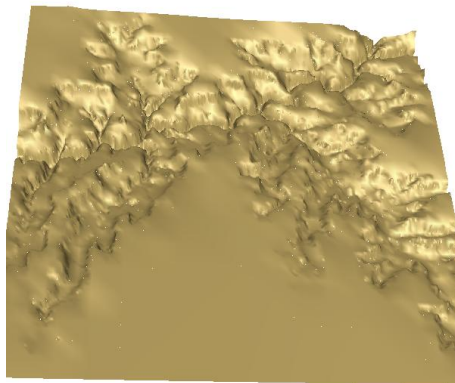
(a) Triangulation de départ.



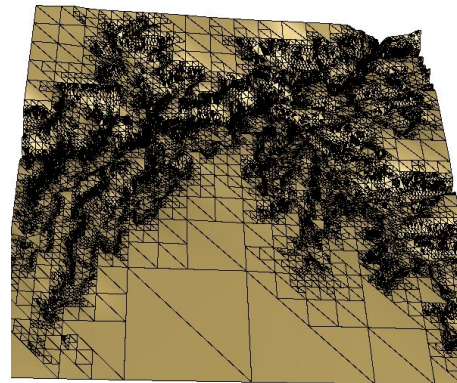
(b) Test avec deux crêtes.



(c) Test avec un plateau.



(d) Partie Est du grand Canyon.



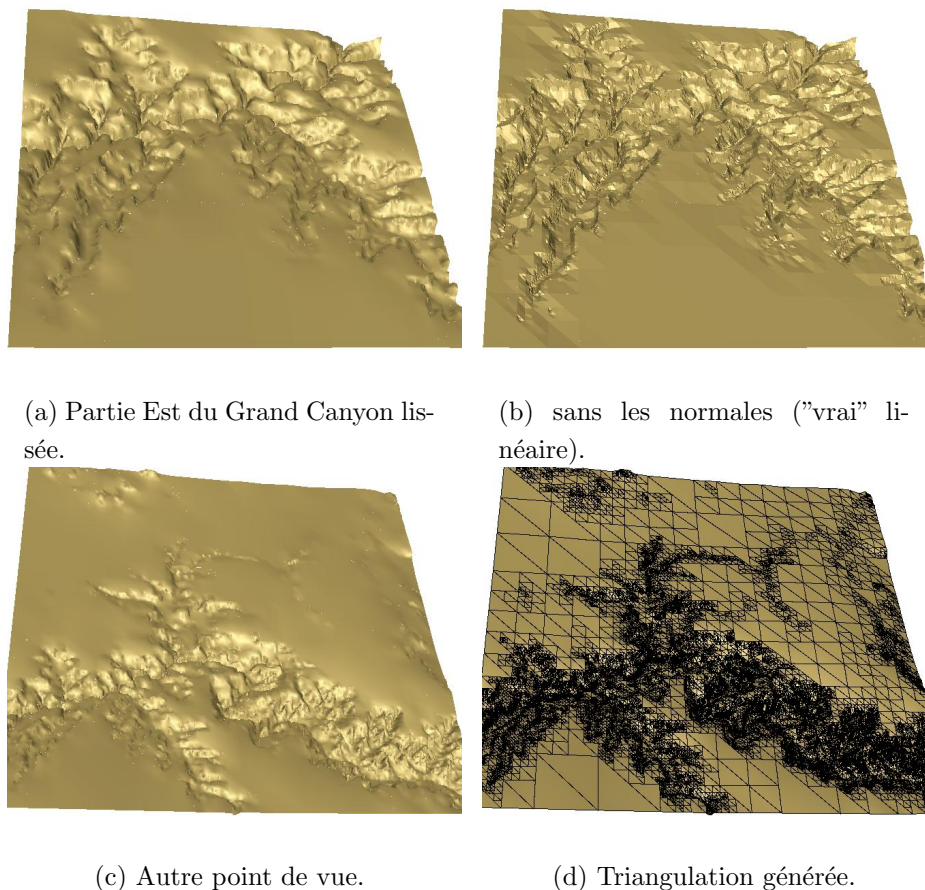
(e) Triangulation engendrée.

**Fig. 3.2.1** – Reconstruction hiérarchique linéaire.

étant séparé de trois secondes d'arc du suivant (c'est-à-dire environ 100 mètres). Deux faces représentant un carré servent de triangulation de base. Pour chaque face, l'erreur en norme infinie est calculée entre la face et les points du maillage. En pratique, les points du tableau appartenant à la face (en projection verticale) sont parcourus puis la valeur absolue de la différence entre l'altitude donnée par le tableau et l'altitude du point de la face est calculée. Si cette erreur est supérieure à une erreur maximale autorisée, l'une des arêtes de la face est subdivisée. Le point libre ainsi créé est déplacé pour coïncider avec celui du tableau. Et ainsi de suite,

toutes les faces sont traitées par niveau (grâce à un parcours en largeur de l'arbre des faces). Une partie des Etats-Unis en relief est ainsi obtenue (figure 3.2.1).

La surface est linéaire par morceaux ce qui n'est pas très agréable à regarder. Un rendu ombré est utilisé pour lisser la surface affichée (voir figure 3.2.2).



(a) Partie Est du Grand Canyon lissée.

(b) sans les normales ("vrai" linéaire).

(c) Autre point de vue.

(d) Triangulation générée.

**Fig. 3.2.2** – Reconstruction du grand canyon.

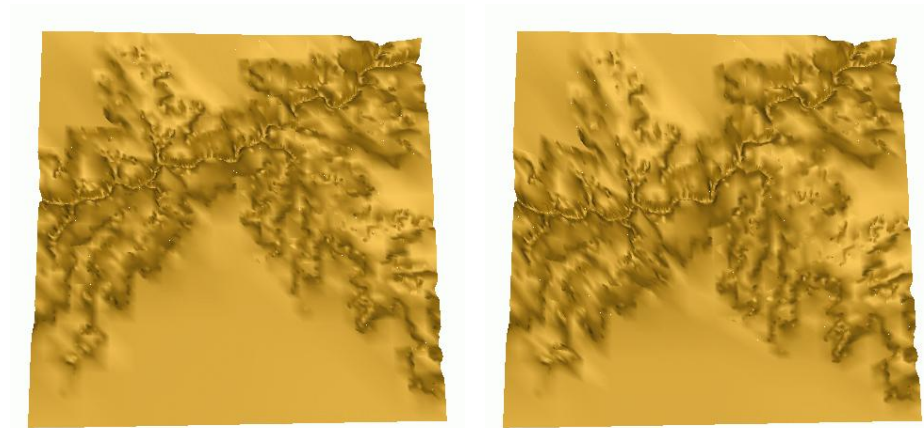
La méthode permet notamment de faire de la compression. En effet, aux endroits relativement plats, seuls de grands triangles sont créés sans ajouter d'erreur. Lors de forts changements d'altitude, la triangulation est raffinée pour permettre de coller aux données.

Une fois que cette triangulation obtenue, elle peut être éditée en déplaçant des points. Le cours du Colorado est ainsi modifié en ne bougeant seulement que deux sommets (voir figure 3.2.3).

### Reconstruction à partir d'un ensemble de points

Les données représentent maintenant une tête de statue chinoise. A partir des coordonnées de points de la tête, une grille régulière en coordonnées cylindriques est





(a) Grand Canyon actuel.

(b) Colorado déplacé.

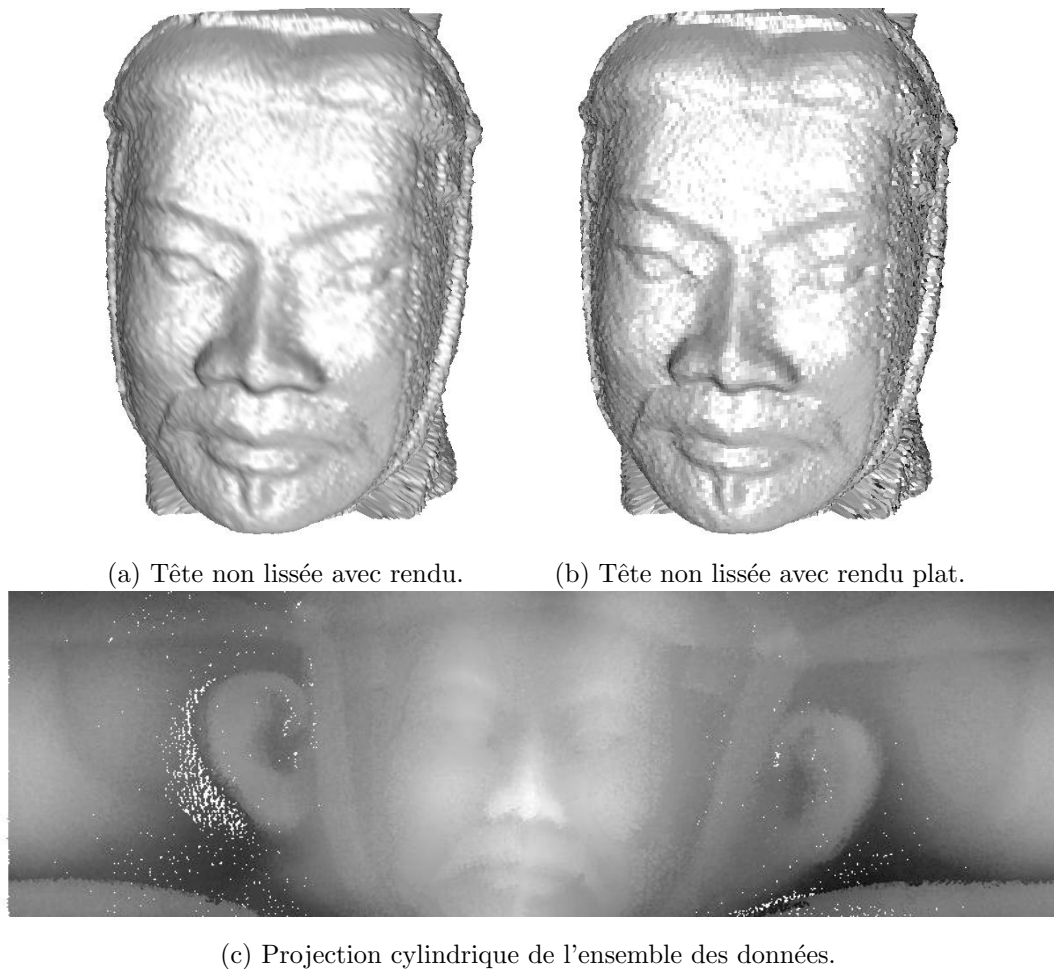
**Fig. 3.2.3** – Déplacement du grand canyon.

générée. Les points de départ étant très nombreux, chaque sommet de la grille régulière est la moyenne de l'ensemble des points se projetant sur lui. Malheureusement, même ainsi, les données sont fortement bruitées ainsi qu'illustré ci dessous en figure 3.2.1. L'ensemble des données présente beaucoup de bruit. Un lissage (filtre par ligne) est appliqué pour réduire le phénomène de bosselage. Chaque point devient le barycentre pondéré de lui-même avec ses voisins. Un filtre 1 2 6 2 1 est utilisé, ce qui signifie que le point  $P_i$  devient :

$$P_i = \frac{P_{i-2} + 2P_{i-1} + 6P_i + 2P_{i+1} + P_{i+2}}{12}$$

Les nouvelles données sont nettement meilleures (voir figure 3.2.5). Le principe de calcul d'erreur et de subdivision en cas d'erreur trop importante reste le même. Néanmoins, une projection cylindrique des sommets permet de leur faire correspondre un point de la grille. De légères améliorations ont été apportées car les brusques sauts de rayon des épaules ou du chapeau engendraient beaucoup de subdivisions. Le calcul de l'erreur est modifié en prenant une interpolation bilinéaire des quatres sommets de la grille régulière délimitant la case dans laquelle se trouvait le point de calcul de l'erreur. Cette amélioration diminue de plus de dix pour cent le nombre de faces créées.

Les triangles sont énormément subdivisés. Il n'y a que peu de grands triangles ; c'est normal car des triangles plats sont utilisés pour représenter une surface cylindrique. Il aurait fallu utiliser des triangles cylindriques pour optimiser le nombre de triangles. Néanmoins, ce paragraphe étant un prélude à la hiérarchisation de la méthode  $G^1$ , cela justifie l'intérêt d'intégrer des surfaces lisses à cette hiérarchisation.

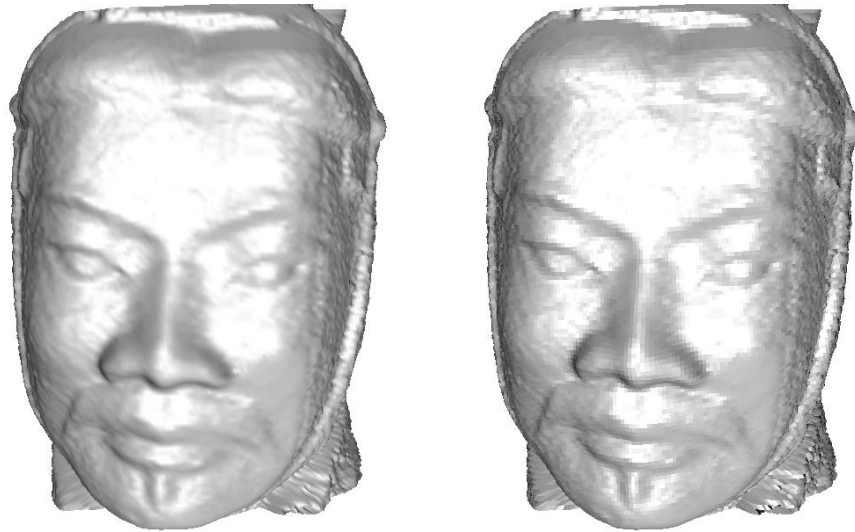


**Fig. 3.2.4** – Données de la tête chinoise avant lissage.

### 3.2.2 Reconstruction de surfaces lisses

La reconstruction d'un objet est souvent découpée en deux étapes : une phase de paramétrisation suivie d'une phase de reconstruction proprement dite. Certaines méthodes ([Krishnamurthy et Levoy, 1996]) combinent ces étapes en générant une paramétrisation simpliste au début puis en itérant optimisation - relaxation de la paramétrisation. Bien que cela semble plus pratique, il paraît plus profitable d'initialiser ces boucles avec une paramétrisation déjà correcte. Rien n'empêche alors d'effectuer une étape de relaxation ultérieurement si nécessaire.

L'objectif est d'obtenir à partir d'un maillage fin une surface multirésolution qui l'interpole. Cet objectif est atteint à travers plusieurs étapes intermédiaires. Tout d'abord, un maillage grossier est déduit du maillage fin. Ensuite, une surface interpolant ce maillage grossier est construite. Comme cette surface est contrôlée par des



(a) Tête lissée avec rendu.

(b) Tête lissée avec rendu plat.



(c) Projection cylindrique de l'ensemble des données lissées.

**Fig. 3.2.5** – Données de la tête chinoise après lissage.

paramètres libres, ceux-ci sont calculés de manière à coller au mieux aux données, c'est-à-dire à minimiser une énergie représentant une erreur. Comme notre surface est paramétrique, les données doivent être elles aussi paramétrées pour pouvoir facilement calculer une erreur. En effet, c'est cette erreur qui est minimisée en fonction des paramètres libres de la surface. Il convient que l'expression de cette erreur soit connue. Si les données ne sont pas paramétrées, l'erreur est définie grâce à une minimisation : pour chaque point du maillage fin à reconstruire, elle est définie comme la somme des distances entre ce point et la surface, c'est-à-dire le minimum des distances entre ce point et tous les points de la surface. Pour la réduire, on est donc confronté à des minimisations imbriquées, lourdes en terme de calcul. Cette paramétrisation obtenue, l'erreur est connue en fonction des paramètres libres de la surface. Les paramètres libres sont alors optimisés par rapport aux données en minimisant cette erreur. Les détails sont finalement ajoutés couche par couche pour



(a) Tête avec rendu erreur 0.1 centimètre.



(b) Tête en flatshading.



(c) Subdivision pour une erreur de 0.1 centimètre.

**Fig. 3.2.6** – Reconstruction de la tête chinoise lissée.

réduire l'erreur à l'aide de raffinements locaux et d'optimisations successives .

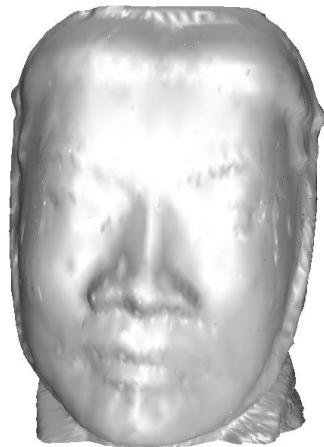
Deux modèles différents ont été utilisés pour illustrer ci-dessous la reconstruction hiérarchique, le lapin de Stanford\* et la tête de Max Planck†.

Cette méthode ainsi que les résultats ont été publiés dans [Yvart *et al.*, 2004b].

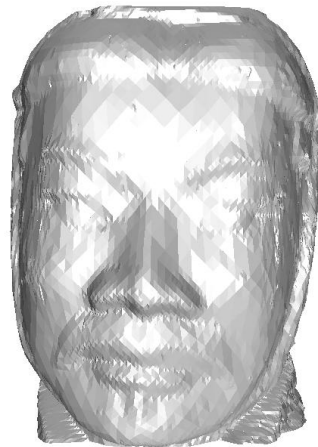
---

\*Merci au Stanford Computer Graphics Laboratory

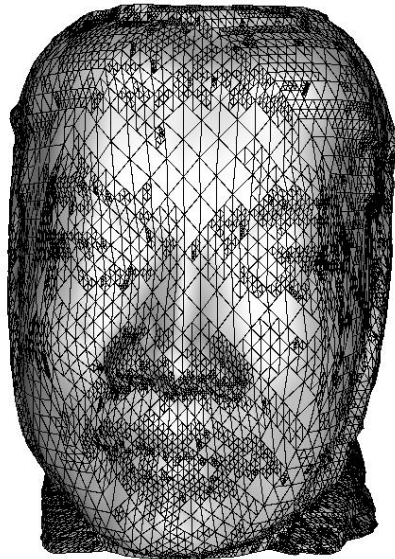
†Merci au Max-Planck-Institut für Informatik



(a) Tête avec rendu erreur 0.3 centimètre.



(b) Tête en flatshading.



(c) Subdivision pour une erreur de 0.3 centimètre.

**Fig. 3.2.7** – Reconstruction de la tête chinoise lissée.

### Calcul d'une paramétrisation

Dans le but de reconstruire un modèle avec notre schéma hiérarchique paramétrique, les données doivent être paramétrées sur un maillage grossier. Les données se présentent sous la forme d'un maillage très fin  $\mathcal{M}$  muni d'un nombre de sommets  $n$ . Pour paramétrer ces données sur un maillage grossier, il faut tout d'abord construire ce maillage grossier  $\mathcal{M}_0$  muni d'un nombre de sommets  $n_0$  très faible. Ce maillage construit, il reste à paramétrer  $\mathcal{M}$  sur  $\mathcal{M}_0$ , c'est-à-dire qu'il faut construire une fonction  $f$  telle que  $f : \mathcal{M} \rightarrow \mathcal{M}_0$ .

Deux types de méthodes s'affrontent pour réaliser ce type de paramétrisation : les méthodes utilisant une procédure par étapes qui permettent à l'utilisateur d'intervenir facilement au cours de la paramétrisation, et les méthodes entièrement automatiques qui génèrent une paramétrisation directement à partir de la simplification de maillage engendrant  $\mathcal{M}_0$ . [Lee *et al.*, 1998] appartient à la seconde catégorie et génère une paramétrisation d'un maillage en paramétrant les simplifications locales (suppressions de sommets) successives. Chaque sommet supprimé est paramétré en fonction de ses voisins directs. A la fin des suppressions successives, l'ensemble des sommets supprimés est paramétré en fonction des sommets restants : on a paramétré le maillage fin sur le grossier. Bien que cette procédure soit élégante, elle rend difficile l'intervention de l'utilisateur. En effet, l'utilisateur connaît l'utilisation future de son modèle. Il peut donc être intéressant pour lui de placer certains sommets spécifiques de la triangulation grossière, ainsi que des arêtes particulières. Cette méthode autorise ce type d'opérations mais le résultat n'est visible qu'en fin de processus. Il faut alors tout recommencer si celui-ci n'est pas satisfaisant. A l'opposé, les méthodes progressives telles [Eck *et al.*, 1995] permettent à l'utilisateur de contrôler chaque étape. Tout d'abord le maillage grossier est construit, ensuite (ou simultanément) le maillage fin est partitionné c'est-à-dire que chaque sommet est associé à une face de la triangulation grossière. Chaque morceau du maillage fin est alors paramétré sur la face de la triangulation grossière qui lui correspond. Enfin le maillage fin peut-être ré-échantillonné si nécessaire. Cette méthode, de par son découpage en étapes simples est plus facile à mettre en œuvre et permet un contrôle étape par étape de la part de l'utilisateur. Bien qu'elle prenne plus de temps pour reconstruire un objet, chose qui n'arrive qu'une fois par objet, elle s'avère plus profitable car l'utilisateur contrôle bien mieux le résultat. Il peut placer tout d'abord les sommets du maillage grossier librement et judicieusement en vue d'une édition hiérarchique, ensuite les courbes de bord pour séparer la surface en morceaux significatifs, et enfin choisir la fonction de paramétrisation pour privilégier la répartition de l'aire ou des angles. La perte de temps est alors compensée par un très fort gain lors de l'utilisation. C'est donc cette seconde méthode que nous avons adaptée pour la paramétrisation d'objets en vue de leur reconstruction.

**Maillage grossier** Pour construire le maillage grossier, un algorithme de décimation de maillage est utilisé, QSlim (voir [Garland, 1998]). Cet algorithme fonctionne sur le principe de la suppression itérative de sommets. Pour chaque sommet, l'erreur induite par sa suppression est calculée. La suppression engendrant la plus petite erreur est effectuée. On recommence ensuite jusqu'à obtention de l'erreur maximum

autorisée ou du nombre voulu de faces.

Comme la surface construite est interpolante, les sommets du maillage grossier constituent un sous-ensemble des sommets des données. Le problème principal est de choisir leur nombre et leur position. L'ensemble des sommets doit être petit dans le but d'exploiter au mieux le principe de niveau de détail, mais il est intéressant en vue de l'édition hiérarchique globale de conserver certains aspects globaux avec des points judicieusement placés. Il convient de réaliser que si l'on ne peut pas interpoler directement le maillage constitutif des données à cause de sa taille (pouvant être de dizaines de millions de faces), il est possible d'interpoler des maillages de plusieurs milliers de faces et plus. L'utilisateur reste donc libre de son choix et pourrait décider de ne prendre que 10 points comme il pourrait choisir 3549. Le facteur déterminant est plutôt l'utilisation prévue, et la sémantique associée. Le plus pratique est de fixer un sommet par ensemble à éditer ultérieurement. Le nombre initial de sommets constitue un compromis entre la facilité d'utilisation et la légèreté de représentation.

Les maillages de base du lapin et de Max Planck sont présentés en figures 3.2.17(a) et 3.2.18(a) avec respectivement 60 sommets et 110 faces pour le lapin et 40 sommets et 70 faces pour Max Planck. L'utilisateur peut aussi imposer certains sommets en vue d'une édition particulière.

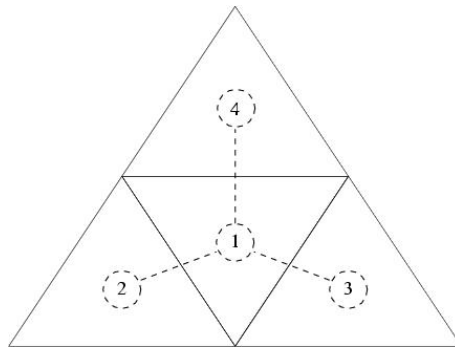
D'un point de vue algorithmique, cette étape présente un coût en  $\mathcal{O}(n)$  avec  $n$  nombre de sommets du maillage fin. En effet, il fonctionne grâce à une file avec priorité : chaque sommet est inséré avec comme priorité l'erreur induite par sa suppression (coût  $\mathcal{O}(n)$ ). Après cela, il reste à supprimer le sommet de priorité minimale (coût  $\mathcal{O}(1)$ ). Les priorités des sommets voisins sont ensuite mises à jour, et on recommence jusqu'à obtention d'un nombre de faces fixé. Globalement, le coût est en  $\mathcal{O}(n)$  ce qui assure de pouvoir l'utiliser même avec des maillages très gros.

**Courbes frontières** Après la création de ce maillage de base, le maillage fin est paramétré sur ce maillage grossier. La première chose faite est de découper le maillage fin en parties correspondant aux faces du maillage de base. Pour cela, l'image de chaque arête du maillage de base est définie comme une courbe sur le maillage fin. La partie de maillage fin incluse entre les images des trois arêtes d'une face définit l'image de la face. Les extrémités de ces courbes sont connues car les sommets du maillage grossier appartiennent au maillage fin. Le reste des courbes est calculé de manière à relier ces sommets par des courbes proches des géodésiques. Cela se fait en deux étapes :

1. Un chemin de longueur minimale sur le graphe défini par le maillage fin est calculé,

2. Ces courbes sont tendues en supprimant itérativement les angles locaux (voir [Bonneau et Hahmann, 2003]).

**Chemin de longueur minimale** On définit un graphe à partir du maillage fin : une face du maillage est un sommet, une arête entre deux faces du maillage correspond à une arête entre deux sommets du graphe (voir figure 3.2.8). Ces arêtes



**Fig. 3.2.8** – Création du graphe à partir du maillage.

se voient affectées de la longueur à parcourir sur le maillage pour rejoindre les barycentres des deux faces du maillage en passant par le milieu de leur arête commune. Grâce à cette opération, on obtient un graphe sur lequel la théorie des graphes s'applique. On peut alors calculer un plus court chemin entre deux sommets, c'est-à-dire entre deux faces du maillage. Ce plus court chemin sur le graphe est un chemin de faces sur le maillage. On construit alors les courbes en reliant les milieux des arêtes communes à ces faces.



**Fig. 3.2.9** – Chemins de longueur minimale.

Le coût algorithmique de cette étape est dans le pire des cas  $\mathcal{O}(n)$  par arête. En fait, le nombre de sommets visités est de l'ordre de  $n/N_f$  avec  $N_f$  nombre de faces du maillage grossier. Le coût globale de cette étape pour toutes les arêtes est



donc :  $N_{arêtes} \cdot \mathcal{O}(n/N_f) = \mathcal{O}(n)$  où  $N_{arêtes}$  est le nombre d'arêtes de la triangulation grossière.

Ces courbes sont montrées figure 3.2.9. On constate qu'elles contiennent de nombreux angles. Cela est dû à plusieurs aspects : tout d'abord on ne mesure pas la distance réelle entre deux barycentres sur le maillage puisqu'on passe par le milieu de l'arête commune. Enfin les courbes les plus courtes entre deux points du maillage n'ont aucune raison de passer par les barycentres des faces. C'est cette seconde approximation qui génère des angles gênants. La première n'est qu'une simplification sans conséquence. Ces courbes ne nous conviennent pas du fait de leurs angles. En effet, elles sont reconstruites par des courbes polynomiales, donc lisses, ce qui a pour conséquence de générer une erreur trop importante à proximité des sommets des angles. Une deuxième étape s'avère nécessaire : un lissage.

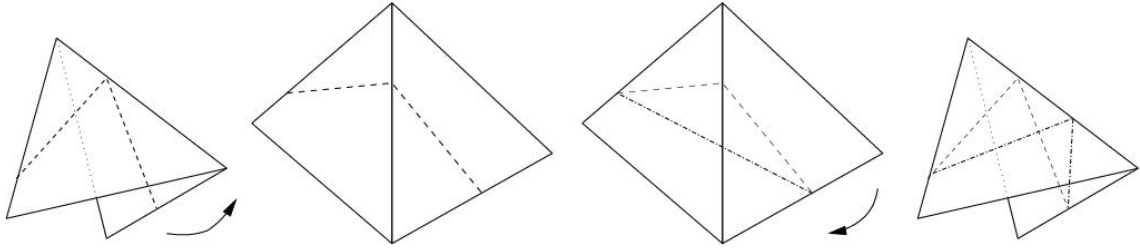
**Lissage** Les courbes que nous venons de calculer présentent des angles fréquents. Il faut les « tendre ». Pour cela, on supprime itérativement les angles en commençant par les plus marqués. L'idée est de considérer deux faces adjacentes le long de la polyligne, et de les déplier de manière à les aplanir. Les deux segments de polyligne deviennent deux segments présentant un angle dans le plan. Ils sont remplacés par un segment de droite reliant les extrémités. Les faces sont ensuite à nouveau repliées (voir les étapes sur la figure 3.2.11). Cette opération est ensuite appliquée à l'angle le plus marqué suivant (après une mise à jour pour tenir compte du changement introduit). Les courbes tendues sont montrées figure 3.2.10. On constate



**Fig. 3.2.10** – Courbes tendues.

que cette étape a bien eu l'effet attendu.

Algorithmiquement, le coût de cette étape est  $\mathcal{O}(n_c)$  pour chaque arête, avec  $n_c$  le nombre de sommets de la courbe de bords. En effet, une file avec priorité est à nouveau utilisée pour laquelle chaque calcul élémentaire est en  $\mathcal{O}(1)$ . Le nombre d'itérations nécessaires est inférieur à 100 dans la pratique. Le coût pour toutes les



**Fig. 3.2.11** – Lissage de la polyligne. Les faces sont dépliées, la polyligne est remplacée par un segment, puis on replie. Localement, la longueur de la polyligne est minimisée.

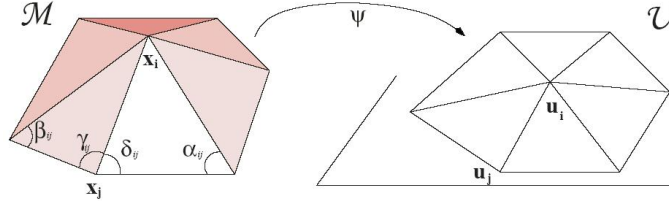
arêtes devient :  $N_{arêtes} \cdot \mathcal{O}(n_c) \ll \mathcal{O}(n)$ .

Lorsque toutes les courbes sont construites, les morceaux du maillage fin correspondant à chaque face sont extraits. Le cas particulier des courbes correspondant aux arêtes de bord du maillage grossier n'est pas développé. En effet deux cas sont possibles : soit elles ne sont pas distinguées et le même traitement que si dessus est appliqué. Cela coupe le maillage fin avec des courbes sans tenir compte des bords du maillage fin. soit le cas des bords est traité à part pour qu'à une arête de bord de la triangulation grossière corresponde la polyligne constituée des arêtes de bord de la triangulation fine. C'est un cas particulier dans l'implémentation qui ne nécessite aucun calcul. Chaque morceau du maillage fin est maintenant paramétré sur la face qui lui correspond.

**Paramétrisation** Chaque morceau du maillage fin est paramétré sur un triangle unitaire, c'est-à-dire à chaque sommet du maillage est associé un couple de paramètre dans le triangle unitaire. Les opérations suivantes sont donc effectuées pour chaque face de la triangulation grossière. Les bords sont fixés de manière évidente : les paramètres sont répartis sur le bord du triangle en fonction de l'abscisse curviligne des sommets sur les courbes de coupe. L'idée est maintenant d'écrire tous les paramètres des sommets intérieurs en fonction de ces bords. Une résolution d'un système d'équations permettra de trouver les valeurs des paramètres correspondantes. [Desbrun *et al.*, 2002] a démontré que l'ensemble des paramétrisations était un espace vectoriel de dimension deux. En effet, une paramétrisation est obtenue par minimisation d'une énergie  $E$ , qui s'écrit sous la forme :

$$E = \lambda E_\Lambda + \mu E_\chi$$

Les variables  $\lambda$  et  $\mu$  permettent de pondérer la conservation de l'aire ou des angles



**Fig. 3.2.12** – Paramétrisation autour d'un sommet et définition des angles - Image extraite de [Desbrun *et al.*, 2002].

entre les différents triangles par rapport aux faces de la triangulation. La minimisation de ces énergies entraîne la résolution d'un système linéaire qui permet de calculer les  $U^i$ , valeurs des paramètres des sommets intérieurs, en fonction des  $C^b = U^b$ , valeurs des paramètres des sommets de bord :

$$MU = \begin{bmatrix} \lambda M^\Lambda + \mu M^\chi \\ 0 & I \end{bmatrix} \begin{bmatrix} U^i \\ U^b \end{bmatrix} = \begin{bmatrix} 0 \\ C^b \end{bmatrix} = C \quad (3.2.1)$$

où  $M^\Lambda$  et  $M^\chi$  sont définis par :

$$M_{i,j}^\Lambda = \begin{cases} \cot(\alpha_{i,j}) + \cot(\beta_{i,j}) & \text{si les sommets } i \text{ et } j \text{ sont voisins,} \\ -\sum_{k \text{ voisin de } i} M_{i,k}^\Lambda & \text{si } i = j \\ 0 & \text{sinon,} \end{cases} \quad (3.2.2)$$

$$M_{i,j}^\chi = \begin{cases} \cot(\gamma_{i,j}) + \cot(\delta_{i,j}) & \text{si les sommets } i \text{ et } j \text{ sont voisins,} \\ -\sum_{k \text{ voisin de } i} M_{i,k}^\chi & \text{si } i = j \\ 0 & \text{sinon,} \end{cases} \quad (3.2.3)$$

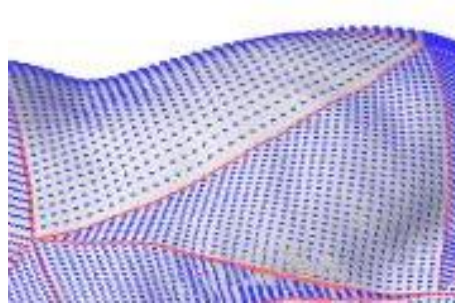
les angles  $\alpha_{i,j}$ ,  $\beta_{i,j}$ ,  $\gamma_{i,j}$  et  $\delta_{i,j}$  sont donnés par la figure 3.2.12.

Il reste à choisir  $\lambda$  et  $\mu$  en favorisant les angles ou les aires. Dans les exemples montrés plus bas, une paramétrisation conservant principalement les proportions des aires nous a semblé plus adaptée à nos modèles.

D'un point de vue algorithmique, cette étape est la plus lourde car il s'agit d'inverser une matrice de taille le nombre de sommets du maillage fin dans une face du maillage grossier, i.e. le nombre  $n$  de sommets du maillage fin divisé par le nombre de faces  $N_f$  du maillage grossier :  $n/N_f$  (asymptotiquement). Néanmoins cette matrice est creuse, chaque ligne contient autant d'éléments non nuls que le sommet considéré admet de voisins : approximativement entre 2 et 10. L'inversion de cette matrice par des méthodes itératives s'effectue donc avec un coût en  $\mathcal{O}(n/N_f)^2$  par face du maillage grossier, soit  $\mathcal{O}(n^2/N_f)$

Arrivé à cette étape, le maillage fin est découpé en morceaux dont chacun a été paramétré sur une face de la triangulation grossière. C'est-à-dire qu'à tout point de la triangulation fine correspond un point du maillage grossier. Le maillage fin est donc paramétré sur le maillage grossier.

**Echantillonnage** Une fois l'ensemble du maillage fin paramétré, des points régulièrement répartis en sont extraits. Pour cela, un ensemble de points uniformément répartis dans l'espace sont construits pour chaque face de la triangulation grossière. Leur pré-image par la paramétrisation est alors calculée. Cela fournit un échantillon uniforme du maillage fin. Bien que cette étape ne soit pas nécessaire (on peut se contenter de la paramétrisation du maillage fin qui nous donne l'équivalent d'un échantillon non uniforme), elle simplifie les calculs en fixant les matrices utilisées pour l'optimisation ultérieure. En effet, l'optimisation de la surface reconstruite s'effectue à travers une minimisation de la forme  $\min \|Ax - b\|^2$  pour chaque face,  $x$  vecteur des points de contrôle de la surface et  $b$  données à reconstruire. La matrice  $A$  dépend de la paramétrisation des données, et est donc différente pour chaque face sans un échantillonnage uniforme pour toutes les faces.

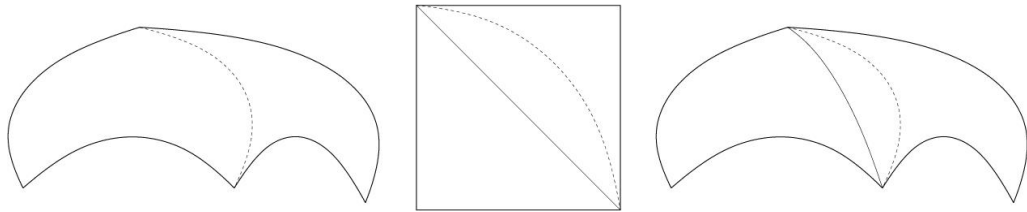


**Fig. 3.2.13** – Echantillon uniforme.

Néanmoins, paramétrer chaque morceau indépendamment ne suffit pas. La figure 3.2.13 montre un échantillon uniforme généré après cette paramétrisation. On constate que chaque échantillon pris séparément est correct, mais pris dans son ensemble, il présente des disparités. La densité des échantillons semble varier entre les patches. Cela est dû aux paramétrisations indépendantes effectuées. Cela signifie que de grandes faces sont voisines de faces plus petites. Pour exploiter au mieux le principe des niveaux de détail, il est plus intéressant d'associer une taille de support à un niveau, et donc d'avoir des densités comparables. Une étape supplémentaire est insérée.

Le coût algorithmique de cette étape provient de la recherche de l'image de la face

fine pour chaque coordonnées  $(u, v)$ . En pratique, cette recherche est effectuée par voisinage et a donc un coût négligeable. L'étape coûte alors  $\mathcal{O}(n)$  si l'échantillonnage contient sensiblement autant de points que le maillage lisse de sommets.

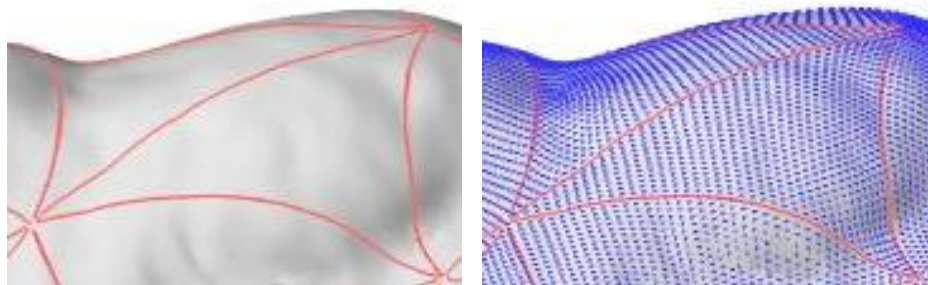


**Fig. 3.2.14** – Lissage de l'échantillon par paramétrisation couplée. Chaque courbe diagonale est remplacée par l'image de la diagonale du carré dans la paramétrisation des deux patches voisins.

**Amélioration de l'échantillon** Ce problème d'hétérogénéité de l'échantillon a été rencontré dans [Eck *et al.*, 1995]. Une solution était de lisser les échantillons entre deux patches adjacents. Pour cela, on paramétrise ces deux patches adjacents sur un carré unitaire. La courbe correspondant à la diagonale est alors remplacée par l'image de la diagonale par la paramétrisation (voir figure section 3.2.14).

Les figures 3.2.15(a) et 3.2.15(b) montrent les courbes et les échantillons uniformes générés après cette étape. On constate que l'échantillon est uniformément réparti même entre les patches. Par ailleurs cette étape améliore la paramétrisation au point qu'il n'est souvent pas nécessaire de faire une étape de correction de celle-ci lors de la reconstruction comme était fait dans [Krishnamurthy et Levoy, 1996] par exemple.

Le coût induit par cette étape est proche de celui des étapes précédentes, mais multiplié par deux ou quatre suivant les étapes car le nombre de points considérés



(a) Courbes lissées

(b) Echantillon uniforme

**Fig. 3.2.15** – Résultats après lissage.

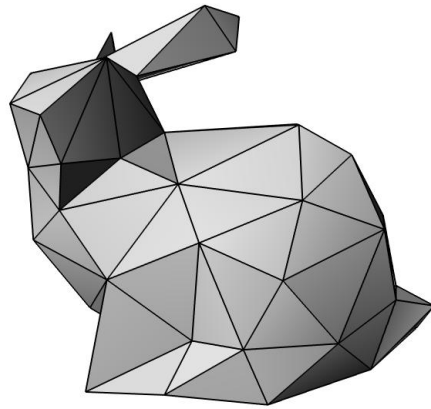
double. Asymptotiquement, il reste le même.

A ce moment du processus, un échantillon uniforme du maillage fin paramétré sur le maillage grossier est obtenu. La figure 3.2.16 montre une vue d'ensemble du modèle détaillé dans les paragraphes ci-dessus. Cet échantillon est composé de ce que nous appellerons des points cibles car ce sont ces points que nous cherchons à atteindre avec une surface hiérarchique.

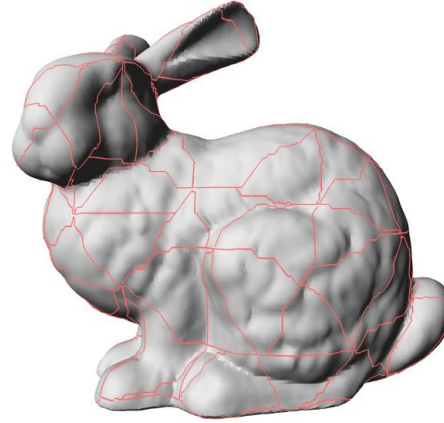
Le coût algorithmique globale est finalement en  $\mathcal{O}(n^2/N_f)$  avec  $n$  nombre de sommets du maillage fin et  $N_f$  nombre de faces du maillage grossier. Ce coût peut sembler prohibitif dans le cas de maillages très gros, mais ce n'est pas le cas car le maillage grossier est généralement adapté au maillage fin. En effet, pour un maillage fin très complexe et muni de nombreuses faces, l'utilisateur construira un maillage grossier capturant quelques aspects globaux, et donc d'une centaine de faces. Le calcul sera alors plus rapide d'un facteur 10 par rapport à un maillage grossier simpliste de dix faces. Par ailleurs, cette opération reste un pré-traitement et ne nécessite pas d'être interactive. Il suffit de ne pas décaler quelques secondes de calcul pour ne pas gêner l'utilisateur, ce qui est le cas dans les exemples cités ci-dessous. De plus, il existe des méthodes d'analyse numérique pour réduire drastiquement le coût d'une inversion de matrice creuse.

### **Reconstruction avec un interpolant hiérarchique**

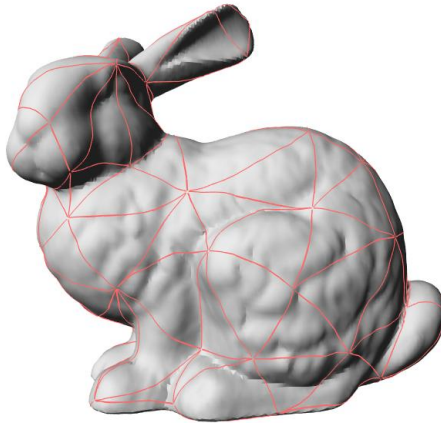
Une fois le maillage grossier et l'échantillon uniforme obtenus, une surface hiérarchique lisse est déterminée pour représenter le maillage très fin donné. Cette surface est calculée selon deux étapes : tout d'abord une surface lisse initiale est construite puis elle est raffinée pour insérer des détails. Cette surface lisse est obtenue par interpolation du maillage grossier. Comme des paramètres libres sont proposés, ils sont optimisés en vue de la reconstruction par une procédure de minimisation d'erreur. L'erreur ainsi minimisée reste trop élevée. Pour la minimiser, la procédure de raffinement reste disponible. En effet, les formes globales des données ont été reproduites par la surface. Mais la surface lisse issue de l'interpolation du maillage initial ne présente pas de détails de taille inférieure aux faces du maillage. Toutes les formes des données de taille plus petite ne peuvent pas être perçues. Le raffinement permet, en diminuant la taille du support de l'interpolant, de prendre en considération ces aspects locaux. Itérativement, toutes les tailles de détail sont traitées. L'erreur de reconstruction finit donc par être très faible.



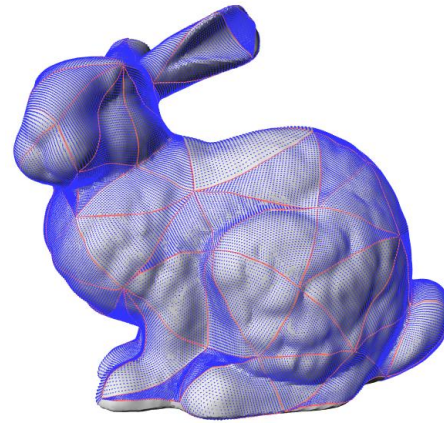
(a) Maillage de base



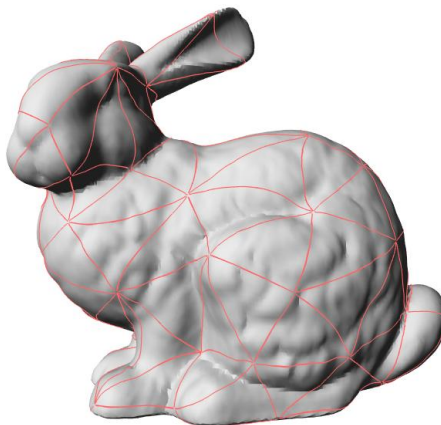
(b) Chemins de longueurs minimum



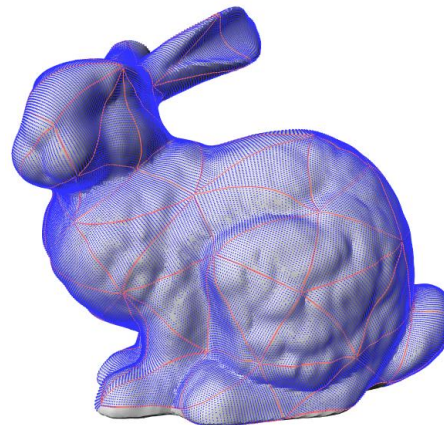
(c) Courbes tendues



(d) Echantillon uniforme



(e) Courbes lissées



(f) Echantillon uniforme

**Fig. 3.2.16** – Paramétrisation du lapin de Stanford.

**Surface initiale** Le maillage grossier sert de base au calcul de la surface lisse initiale. Il suffit maintenant de fixer les paramètres libres pour obtenir une surface. Les paramètres libres sont calculés dans l'ordre : dérivées premières, twists puis points libres intérieurs. Cette décomposition permet à nouveau à l'utilisateur d'intervenir pour adapter le résultat à l'utilisation ultérieure prévue. Il est par exemple envisageable de forcer la convexité des courbes de bord sans se préoccuper des données. Le traitement indépendant des paramètres libres le rend possible. De plus, la surface dépend des dérivées premières de manière non linéaire à cause des fonctions scalaires  $\varphi$ ,  $\mu$  et  $\nu$  introduites dans le chapitre 2.3. Effectuer une minimisation globale de l'erreur sur l'ensemble des paramètres libres simultanément entraînerait une minimisation non linéaire coûteuse et inutile. Il est bien plus intéressant de procéder par étapes en optimisant une partie de la surface à forte dépendance uniquement (ici les courbes de bord). Pour ce faire, l'erreur est réduite à la somme de ses composantes le long des courbes de bord.

Les dérivées premières sont calculées de manière à rapprocher les courbes de bord des polygones leur correspondant. Les dérivées premières sont traitées successivement. L'idée est d'utiliser une minimisation par moindres carrés pondérés (ce qui revient à résoudre un système linéaire) :

$$\min \sum_{i=0}^n \lambda_i \|c(t_i) - p_i\|^2,$$

où  $c(t)$  est la courbe de bord dont les dérivées premières sont en cours de calcul. Les  $p_i$  sont les points cibles le long de cette courbe de bord aux valeurs du paramètre  $t_i$ . Les  $\lambda_i$  sont des poids. Les inconnues sont donc les coefficients du polynôme  $c(t)$  en prenant en compte le fait que les courbes de bord sont des courbes polynomiales de degré cinq par morceaux. Ces coefficients obtenus, les dérivées premières aux extrémités sont déduites. Le coût de ce calcul est extrêmement faible puisqu'il revient à effectuer un moindre carré sur  $n_c$  points, points cibles de la courbe de bord avec deux variables pour chaque coordonnée. Les matrices à inverser sont donc de taille deux. Finalement, le coût est en  $\mathcal{O}(n_c)$

Les polygones limites sont approchés par un polynôme de degré cinq par morceau. Cette approximation est pondérée par des poids plus lourds vers les extrémités. En effet, le milieu sera amélioré ensuite grâce aux raffinements successifs. Les nouveaux sommets étant insérés au milieu des arêtes, localement l'erreur y deviendra nulle. Le polynôme obtenu n'est pas la courbe de bord calculée par l'interpolant puisqu'elle ne prend pas en compte les conditions  $G^1$ . Il n'est pas possible d'effectuer cette minimisation avec la vraie courbe de bord car les twists, et donc les dérivées secondes



ne sont pas encore connues (et ont une influence sur toute la face donc doivent être optimisés plus tard). Néanmoins, on déduit de cette approximation des dérivées premières suffisantes. Ces dérivées premières ne sont pas nécessairement coplanaires, ce qui est impossible pour une surface lisse. Toutefois, si les données sont suffisamment denses, ces vecteurs seront presque coplanaires, une simple moyenne des vecteurs normaux à deux dérivées successives suffit pour obtenir une bonne estimation de la normale à la surface. L'estimation des dérivées est alors projetée dans ce plan pour finir cette étape.

Une fois ces dérivées premières calculées, d'abord les twists puis les points libres intérieurs sont optimisés à leur tour. De nouveau, une optimisation par minimisation aux moindres carrés pondérés est utilisée pour rapprocher les points de la surface des points cibles. Un critère de lissage est rajouté dans le but de prendre une moyenne entre la minimisation de l'erreur d'approximation et l'énergie de lissage. L'énergie minimisée est donc de la forme :

$$E = \lambda E_{approx} + (1 - \lambda) E_{lissage}$$

avec  $\lambda$  paramètre de forme pondérant l'énergie entre l'aspect lisse et l'approximation.  $E_{lissage}$  est une énergie de tension de la surface. La minimiser tend la surface et empêche les ondulations.

Algorithmiquement, le coût est celui de l'évaluation des distances en chaque point cible, i.e.  $\mathcal{O}(n/N_f)$  pour chaque twist. Globalement, pour tous les twists, le coût est donc  $\mathcal{O}(n/N_f * N_e)$  avec  $N_e$  nombre d'arêtes du maillage grossier, c'est-à-dire  $\mathcal{O}(n)$

Il convient ici de remarquer que stricto sensu, l'énergie d'approximation vaut :

$$E_{approx} = \sum_{i=0}^N dist^2(p_i, S)$$

avec  $S$  la surface. L'énergie d'approximation dépend donc de la distance des points cibles à la surface, et non de la distance de chaque point cible au point de la surface de même paramètre. Si le paramètre de  $p_i$  est  $t_i = f^{-1}(p_i)$ ,

$$dist(p_i, S) = \min_t \|p_i - S(t)\| \neq \|p_i - S(t_i)\|$$

L'utilisation de la distance exacte implique donc une minimisation imbriquée, qui est généralement résolue en deux étapes : d'abord les points de contrôle sont optimisés en assimilant la distance à  $\|p_i - S(t_i)\|$ , puis la paramétrisation est corrigée en fixant les points de contrôle et calculant une nouvelle paramétrisation optimale. Cette étape de correction de la paramétrisation est importante dans le cas où la paramétrisation

n'est pas très bonne. Dans notre cas, l'intervention de l'utilisateur et le recalcul des courbes de bord assurent une paramétrisation suffisamment correcte pour ne pas avoir à effectuer cette étape.

Les maillages grossier et les surfaces interpolantes du lapin de Stanford et de Max Planck sont présentées en figure 3.2.17(b) et 3.2.18(b).

**Ajout de détails** Cette surface lisse interpolant le maillage de base ne présente pas tous les détails de l'échantillon. Seul l'aspect global des données ressort. Pour représenter sa précision, une erreur est calculée pour chaque patch de la surface. L'erreur mesurée est le maximum des distances minimum entre les points de l'échantillon (point cible) et la surface lisse. Elle est obtenue en évaluant finement chaque patch de Bézier, puis parcourant localement cette surface autour du paramètre du point cible. Pour réduire l'erreur, la surface peut être raffinée partout pour approcher au mieux l'échantillon, ou localement pour borner l'erreur.

En un sommet nouvellement inséré, la première chose à faire est de changer ses coordonnées en celles du point cible correspondant. Comme le schéma utilisé est interpolant, l'erreur autour de ce sommet sera proche de zéro. La même minimisation qu'au niveau initial est ensuite appliquée localement pour calculer les paramètres libres en ce sommet.

**Raffinement global** Pour obtenir la plus petite erreur possible, chaque arête est subdivisée et les sommets ainsi créés optimisés. Néanmoins, cela engendre de nombreuses faces ( $n \cdot 4^i$ ), et beaucoup d'entre elles sont inutiles car elles ne diminuent pas l'erreur globale si l'interpolation y était suffisamment précise. Les figures 3.2.17(b), 3.2.17(e) et 3.2.17(h) montrent trois niveaux de détail pour le lapin de Stanford ainsi que la répartition de l'erreur. On constate que l'erreur est relativement faible partout sauf dans les zones à fort gradient. L'erreur maximale globale est donc déterminée par l'erreur dans ces zones. Les zones à faible gradient n'interviennent pas dans le calcul de l'erreur maximale. Il était donc inutile d'y subdiviser autant la surface.

**Raffinement local** Pour éviter ces faces inutiles, il suffit de ne subdiviser qu'aux endroits où c'est nécessaire. Pour cela, une limite supérieure est donnée à l'erreur. Si l'erreur sur une face est supérieure à cette borne, chaque arête de cette face est subdivisée. Quand toutes les faces du niveau courant ont été traitées, on s'intéresse aux faces du niveau suivant et ainsi de suite (on effectue donc un parcours

Modèle	Lapin	Tête
#faces	69473	47082
#faces du maillage grossier	110	71
Erreur maximum au niveau 1	4.25 %	3.19 %
Erreur maximum au niveau 2	2.86 %	2.11 %
Erreur maximum au niveau 3	2.10 %	1.12 %
Erreur moyenne au niveau 1	0.39 %	0.44 %
Erreur moyenne au niveau 2	0.17 %	0.21 %
Erreur moyenne au niveau 3	0.07 %	0.09 %

TAB. 3.2.1 – Erreurs mesurées avec les modèles reconstruits.

en largeur de l'arbre des faces). La figure 3.2.19 montre les raffinements locaux pour le lapin de Stanford et la tête de Max Planck.

Algorithmiquement, chaque raffinement induit un calcul autour d'un sommet d'ordre six, le nombre de points cibles autour étant quatre fois plus petit. Néanmoins le nombre de sommets augmente. En fait, chaque point cible est visité une fois pour le calcul de l'erreur en ce point. Le coût globale est donc le même. Pour la première étape, le coût est  $\mathcal{O}(n)$ , un premier raffinement global coûte lui aussi  $\mathcal{O}(n)$  et ainsi de suite. Un raffinement local permet en revanche de profiter de la baisse du nombre de points cibles visités, et est donc beaucoup plus rapide.

## Résultats

Plusieurs reconstructions avec différents niveaux de détails ont été calculées basées sur les modèles du lapin et de la tête de Max Planck, voir figure 3.2.17 et 3.2.18. Les résultats en terme d'erreur et de nombre de faces sont présentés en table 3.2.1. Même les surfaces de base avec 110 et 71 faces sont très précises puisque l'erreur maximum est de moins de 5 %. Les aspects globaux sont reproduits dès le niveau grossier grâce au positionnement des sommets du maillage de base. Cependant, les données sont lissées car la surface est constituée de trop peu de patches, ne pouvant pas reproduire les détails fins. Pour modéliser ces détails, des raffinements sont nécessaires. Deux niveaux de raffinements sont montrés en figure 3.2.17(e) et 3.2.17(h). Cette représentation est performante car l'erreur moyenne est divisée par deux voire plus pour chaque nouveau niveau de détail ajouté. Les détails sont clairement ajoutés dans l'ordre décroissant de leur taille : d'abord les globaux puis les locaux. La figure 3.2.17(e) présente quelques bosses de la fourrure du lapin qui se précisent en figure 3.2.17(h). La figure 3.2.17(k) montre l'erreur d'approximation. Les erreurs

maximales apparaissent localisées autour des parties à fort gradient de la surface. Il est donc intéressant de raffiner localement pour profiter des parties lisses de l'objet et de les représenter avec peu de patches. La figure 3.2.19 présente le résultat de raffinements locaux pour modéliser le lapin et la tête avec une erreur inférieure à 1.5 %. Grâce au raffinement local, le nombre de faces est de 653 pour le lapin au lieu de  $110 \cdot 4^2 = 1760$ . De même pour la tête de Max Planck, le raffinement local permet de passer de  $71 \cdot 4^2 = 1136$  à 446. La tête de Max Planck profite particulièrement de ces raffinements locaux avec de grands patches sur son front et des fins autour des zones détaillées comme les yeux par exemple. Par ailleurs, les résultats présentés ici concernent l'erreur maximale ou moyenne, mais il est intéressant de regarder la répartition de l'erreur (voir figures 3.2.20 et 3.2.21). Non seulement l'erreur maximale diminue, mais la répartition indique que les erreurs les plus grandes sont de moins en moins nombreuses.

Les temps de calculs rencontrés sont de l'ordre de quelques secondes pour la génération de l'échantillon, et moins d'une seconde pour l'optimisation de la surface. Il ressort que ces délais sont presque transparents pour l'utilisateur. Le coût global de la reconstruction est fixée par la paramétrisation de chaque morceau du maillage fin sur un triangle unitaire. Cette étape constitue le goulot d'étranglement de notre méthode, mais les calculs peuvent se faire indépendamment sur plusieurs stations, et les améliorations futures de la paramétrisation sont facilement intégrables.

### 3.2.3 Conclusion

Un objet réel ou déjà modélisé par une autre technique peut être reconstruit pour tirer profit des avantages offerts par notre nouvelle méthode. Pour cela, il suffit de pouvoir l'exporter sous forme de triangulation, chose facile pour un objet déjà numérisé car c'est le principe de l'affichage 3D. Les objets réels sont saisis par un scanner 3D puis prétraités. Une succession d'étapes donne le contrôle à l'utilisateur pour adapter le résultat en vue de l'utilisation prévue. Cette reconstruction effectuée, le modèle hiérarchique lisse est pleinement exploitable en offrant la facilité des niveaux de détails comme l'édition globale quelle que soit la topologie.

Cette procédure, bien qu'elle puisse être utilisée entièrement automatisée, comme [Lee *et al.*, 1998], permet à l'utilisateur un contrôle et une correction étape par étape dans le but d'adapter la reconstruction à l'utilisation future de l'objet. Il faut ici remarquer que la reconstruction s'effectue une unique fois alors que le modèle sera utilisé à de nombreuses reprises. Le temps passé à améliorer le résultat en vue d'un traitement spécifique est donc vite amorti. L'utilisateur pourra choisir de placer des

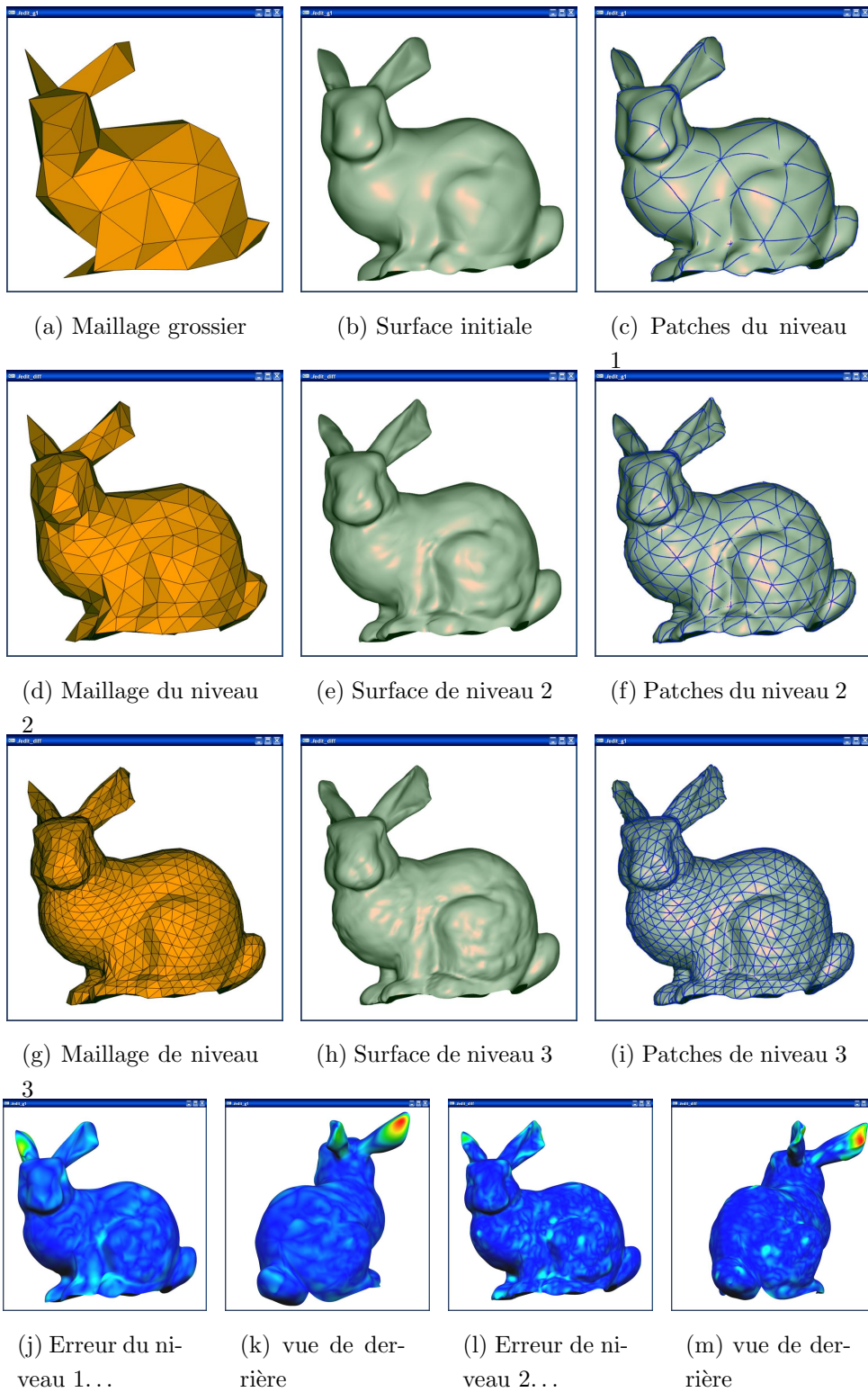
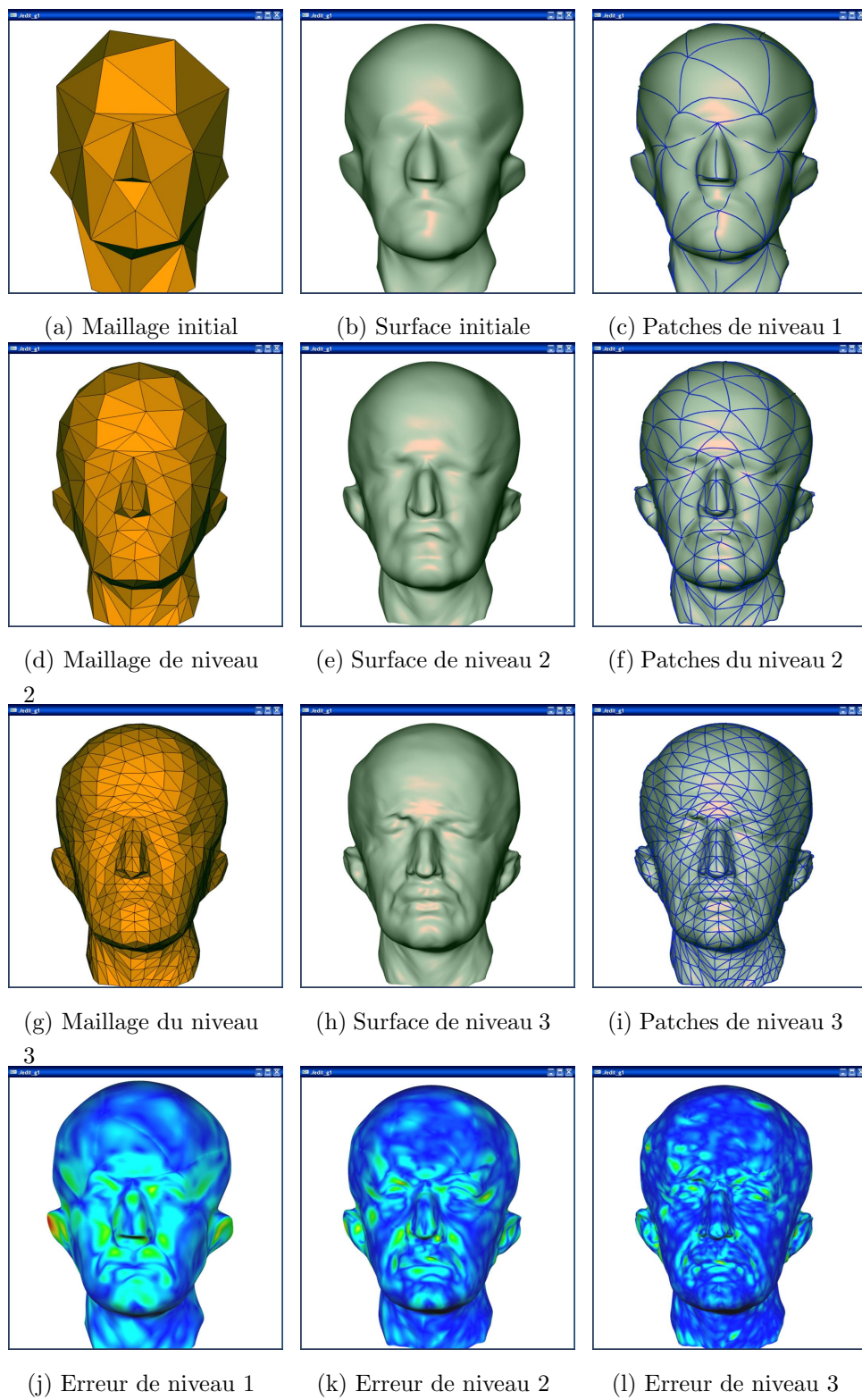
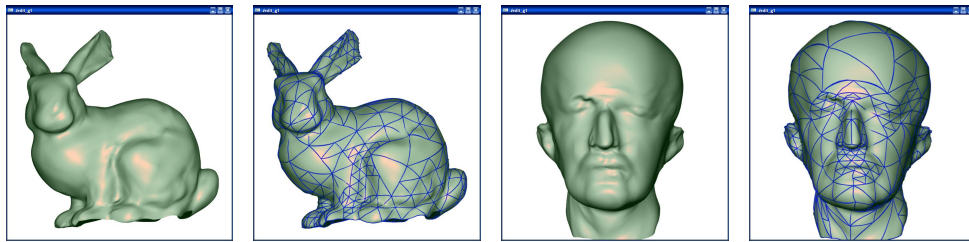


Fig. 3.2.17 – Reconstruction du lapin de Stanford.

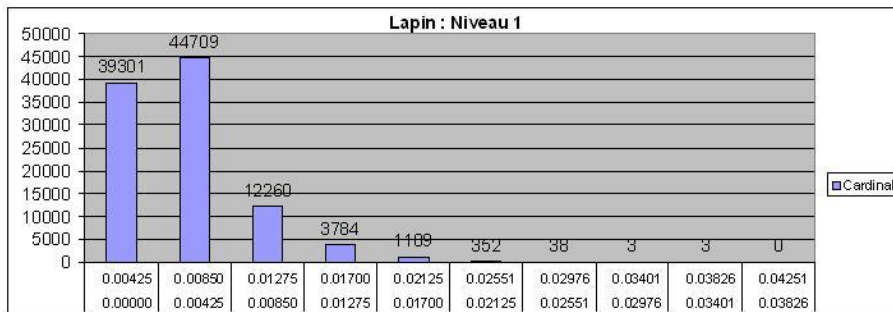


**Fig. 3.2.18** – Reconstruction du buste de Max Planck.

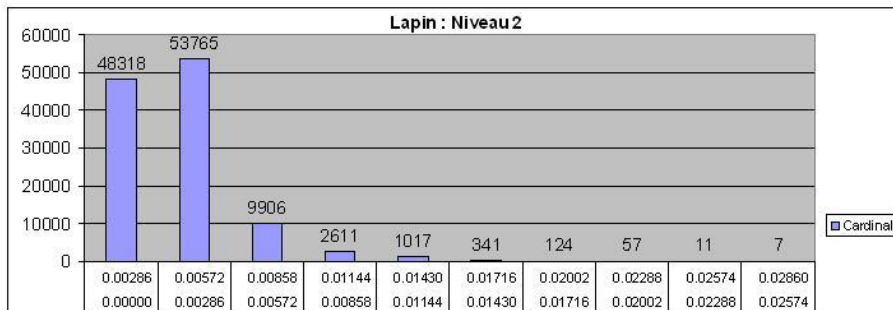


(a) Lapin avec une erreur < 1.5%      (b) 653 patches      (c) Max Planck avec une erreur < 1.5%      (d) 446 patches

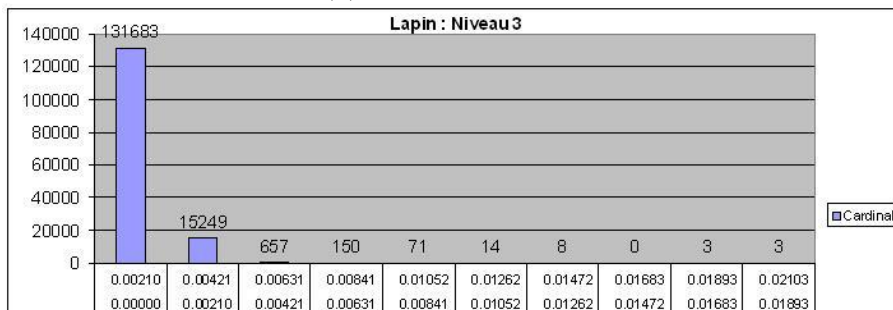
Fig. 3.2.19 – Reconstructions locales.



(a) Lapin au niveau 1

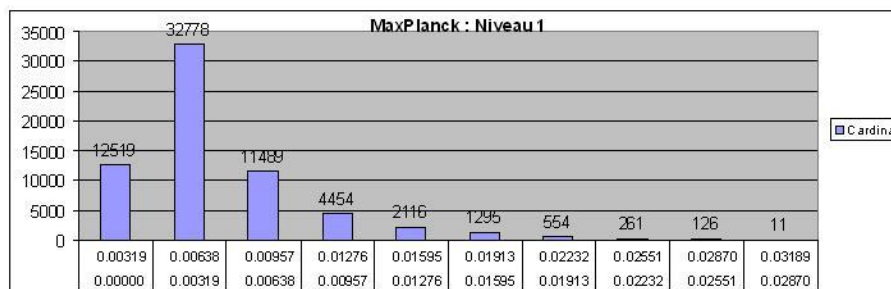


(b) Lapin au niveau 2

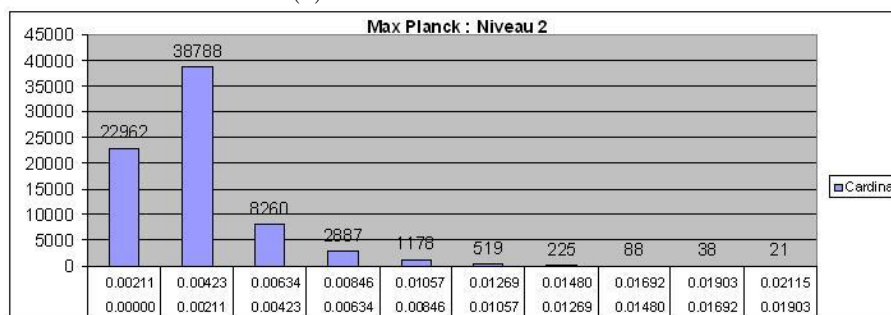


(c) Lapin au niveau 3

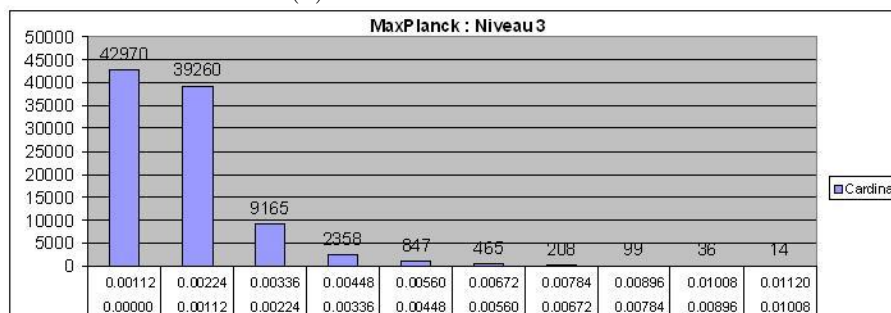
Fig. 3.2.20 – Répartition de l'erreur du lapin : Nombre de points compris entre deux seuils d'erreur.



(a) Max Planck au niveau 1



(b) Max Planck au niveau 2



(c) Max Planck au niveau 3

**Fig. 3.2.21** – Répartition de l'erreur pour Max Planck : Nombre de points compris entre deux seuils d'erreur.



sommets en des endroits spécifiques, changer les critères de construction des courbes reliant ces sommets (actuellement des quasi-géodésiques) ou même en fixer certaines à la main, choisir la méthode de paramétrisation la plus adaptée pour chaque face. Une étape de lissage améliore grandement la paramétrisation en modifiant les courbes de bord. Cette étape n'est pas indispensable mais ne pas l'effectuer implique l'utilisation d'une relaxation de la paramétrisation lors de la reconstruction. Si elle est appliquée, il est possible de spécifier quelles courbes ne doivent pas être modifiées (courbes données par l'utilisateur ou calculées de manière particulière). Tant que les courbes non modifiées ne sont pas excessivement nombreuses, l'étape de lissage reste très utile.

Le calcul et l'optimisation d'une surface à partir des données paramétrées s'effectue par des minimisations aux moindres carrés successives. Là encore, le découpage en étapes permet à l'utilisateur d'intervenir et simplifie les calculs. Auparavant, l'optimisation se faisait sur l'ensemble des points de contrôle en incluant une contrainte de continuité  $G^1$ . Notre méthode générant uniquement des surfaces de continuité  $G^1$ , l'optimisation sur un sous-ensemble de points de contrôle inclut ces contraintes et simplifie les étapes de minimisation.

La méthode proposée s'inspire de [Eck *et al.*, 1995] mais s'adapte aux particularités des surfaces lisses et des paramètres libres offerts. Le découpage du processus en blocs élémentaires permet de bénéficier des avancées spécifiques aux différents domaines abordés (lissage de courbe, paramétrisation, optimisation, etc) et l'automatisation ou non de chacun d'eux. Le grand nombre de paramètres libres de notre méthode et leur séparation dans le calcul offre une grande souplesse d'utilisation. Contrairement au cas des surfaces par ondelettes, il n'est pas nécessaire de ré-échantillonner le maillage. (En pratique, cela n'a été fait que pour des raisons algorithmiques.) L'erreur d'échantillonnage n'existe alors plus, et l'utilisation de surfaces lisses interpolantes rend l'erreur d'approximation très faible quelle que soit la courbure locale du maillage fin. En cas de transmission progressive, l'utilisateur observe immédiatement des surfaces lisses et agréables.

La reconstruction d'objets de type topologique quelconque se révèle facile grâce à la méthode que nous avons présentée. La décimation du maillage fin conserve le type topologique, et les bords ne représentent que des courbes fixées et déterminées à ne pas calculer. La présence de bords allège donc les calculs. Par ailleurs, l'utilisation de surfaces définies sur des triangulations permet de représenter tous les objets possibles.

## Conclusion

Dans la partie 2, une modélisation hiérarchique de surfaces lisses a été introduites. Dans le but de la mettre en valeur, deux applications principales ont été développées : un modeleur 3D ainsi qu'un outil de reconstruction. Le modeleur 3D permet à un designer de bénéficier directement des avantages de notre méthode hiérarchique à travers notamment une construction par niveaux de détails. Ce principe est proche de la démarche créatrice d'un artiste qui commence par dresser les lignes d'ensemble avant d'affiner son schéma. Ce modeleur génère des surfaces paramétriques de continuité  $G^1$  manipulables non seulement à l'aide de la hiérarchisation, mais aussi par de nombreux paramètres libres. L'utilisation de maillage sous-jacents triangulaires rend possible la construction d'objets de n'importe quel type topologique. L'outil de reconstruction quant à lui permet de définir un objet déjà existant par des surfaces lisses hiérarchiques pour les introduire dans une scène ou animation et quels que soient le type topologique de l'objet et l'existence de bords.

Par ailleurs, grâce à la représentation par niveaux de détails proposée, un rendu adaptatif est possible pour assurer une fréquence d'affichage minimale et cela quelle que soit la complexité de l'objet traité.

Notre méthode est d'ores et déjà pleinement exploitable dans le cadre de la modélisation géométrique puisqu'elle répond aux exigences récoltées : liberté de création, facilité de représentation et d'édition.



# Conclusion

L'objectif de cette thèse était de développer un modèle de surfaces polynomiales, interpolantes, globalement lisses dans le sens continuité du plan tangent et hiérarchiques (à niveaux de détails) dans le but de faciliter les démarches de création d'objet et d'édition pour un designer.

La demande pour ces propriétés est venue naturellement suite à l'étude des méthodes actuellement existantes. Nous avons déduit de ce tour d'horizon de la modélisation de surfaces les qualités nécessaires à l'utilisation d'une modélisation (à savoir d'être polynomiale, interpolante, locale, globalement lisse et hiérarchique) ainsi que les écueils à éviter.

Ces propriétés décidées et justifiées, il restait à créer une modélisation de surfaces y répondant. La combinaison des propriétés d'être paramétrique et hiérarchique nécessitait un éclaircissement. Nous avons donc étudié comment mettre en place la hiérarchisation pour une surface paramétrique. Celle-ci se fait à travers un raffinement, en l'occurrence une subdivision de l'espace des paramètres en quatre par insertion de sommet en milieu d'arête. Après ces précisions, un schéma initial de construction de surfaces lisses a été détaillé en présentant les modifications à y apporter en vue de sa transformation en méthode hiérarchique. L'impact d'un raffinement sur ce schéma a alors été abordé pour montrer que la surface n'est pratiquement pas modifiée par une subdivision. Après une subdivision, une édition locale permet d'ajouter un détail. Néanmoins comme la construction d'un objet passe par une succession de raffinements, nous avons expliqué en quoi l'utilisation de niveaux de détails pouvait être profitable à travers l'édition hiérarchique qui constitue un outil de déformation globale en conservant les détails locaux. Des exemples de mise en œuvre de la méthode présentée dans cette thèse sont fournis dans le cas d'un objet simple homéomorphe à un tore ou d'un objet complexe

Une fois la méthode de modélisation hiérarchique de surfaces expliquée et testée, nous avons tenu à la mettre en valeur en l'utilisant au sein d'applications. Pour cela, un modeleur 3D ainsi qu'un outil de reconstruction ont été développés. La

partie modelleur était directe puisqu'il s'agissait avant tout de permettre l'accès à l'ensemble des possibilités de la modélisation hiérarchique : les paramètres libres, le raffinement et l'édition. La partie reconstruction nécessitait un travail beaucoup plus approfondi. En effet, la reconstruction d'objet est un sujet dur ayant entraîné beaucoup de travaux, en particulier en vue de son automatisation. Cependant, la reconstruction est une étape unique alors que l'utilisation ultérieure est fréquente. Il est donc rentable de passer du temps à optimiser la surface en vue de son utilisation future pour profiter au maximum des possibilités offertes par la modélisation hiérarchique. Pour cette raison, l'outil développé permet un constant contrôle de l'utilisateur, mais aussi son intervention.

## Travaux futurs

Finalement, nous avons réussi à répondre à la problématique et aux contraintes que nous nous étions imposées. La méthode proposée reprend avantageusement les qualités de plusieurs méthodes existantes. Cela ne conclut pas pour autant les travaux développés dans cet ouvrage.

Plusieurs extensions sont envisageables : tout d'abord la topologie de la surface créée est fixée par le maillage sous-jacent. Or rien ne force cette topologie à être constante. Il est tout à fait acceptable d'imaginer rejoindre deux morceaux de surface (ou creuser un trou) pour créer une anse à n'importe quel niveau de la hiérarchie. La seule limitation provient du fait que la surface raffinée est souhaitée identique. Cette contrainte, bien que naturelle, peut être assouplie. Un morceau de surface pourrait alors être remplacé ou fusionné avec un autre. De la même façon, nous avons travaillé avec des surfaces lisses car la majorité des surfaces rencontrées le sont. Mais il arrive qu'un objet présente localement une arête vive : une crête. Bien que ce cas n'ait pas été traité, il ne pose pas de difficulté majeure puisqu'au contraire, il réduit localement les contraintes sur la surface. Il est facile de calculer deux surfaces lisses ayant un bord commun. Ces extensions sont possibles mais ne présentent qu'une difficulté algorithmique, et ont donc été laissées de côté.

Dans la continuité de ces travaux, il serait intéressant de les appliquer dans d'autres contextes d'utilisation. On peut notamment envisager les domaines de l'imagerie médicale et de la simulation physique. L'une des pistes à explorer serait l'intégration des contraintes physiques dans le modèle géométrique en liant les niveaux de détails par des équations issus de la physique. Ces équations pourraient être aussi variées que : fonctions de conservation de volume ou de surface, déformation élastique

du matériau voire des muscles. . .

## **Au-delà**

Pour terminer notre travail, en réponse aux problématiques rencontrées au cours de cet ouvrage : reproduction du réel ou création artistique, il faut favoriser une démarche « naturelle » basée sur une construction progressive : un maillage grossier (cellule souche) affiné graduellement pour donner naissance à l'objet fini (être vivant).



# **Annexes**





# Annexe A

## Notions de CAGD

### Sommaire

---

<b>A.1</b>	<b>Contuinité géométrique</b>	<b>131</b>
A.1.1	Méthode de Farin	132
A.1.2	Méthode de Chiyokura-Kimura	133

---

### A.1 Contuinité géométrique

Soient  $Q(u, v)$  et  $R(v, w)$  deux facettes surfaciques. Ces deux facettes se raccordent avec continuité géométrique d'ordre un ( $G^1$ ) si elle partagent un même plan tangent continu le long d'une courbe de bord commune  $\Gamma(v) = Q(0, v) = R(v, 0)$ . Mathématiquement, cette propriété peut être écrite sous la forme :

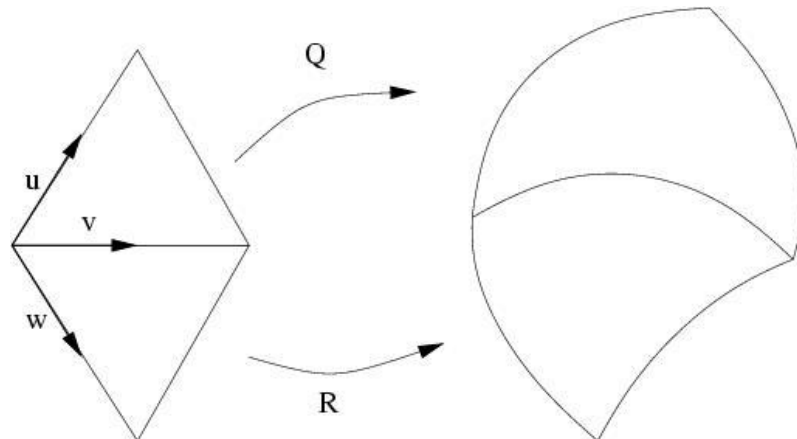


Fig. A.1.1 – Patch  $Q$  et  $R$  partageant une courbe de bord.

**Définition A.1.1**  $Q$  et  $R$  se raccordent avec continuité géométrique d'ordre un noté  $G^1$  si et seulement si les conditions suivantes sont satisfaites :

(C1)  $Q(0, v) = R(v, 0)$ , i.e. raccordement de continuité  $G^0$ ,

(C2) Il existe trois fonctions réelles  $\varphi$ ,  $\mu$  et  $\nu$  telles que

$$\varphi(u) \frac{\partial Q}{\partial v}(0, v) = \mu(v) \frac{\partial Q}{\partial u}(0, v) + \nu(v) \frac{\partial R}{\partial w}(v, 0), \quad (\text{A.1.1})$$

i.e. les trois dérivées premières sont coplanaires,

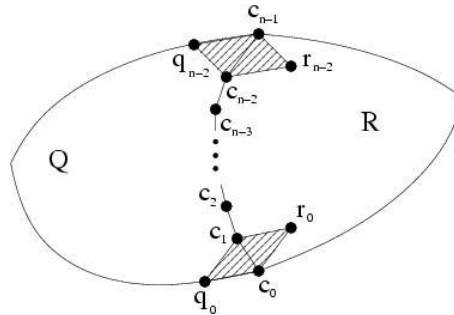
(C3)  $\mu(v)\nu(v) > 0$ , i.e. chaque surface se trouve d'un côté de la courbe de bord commune,

(C4)  $\frac{\partial Q}{\partial v}(0, v) \times \frac{\partial Q}{\partial u}(0, v) \neq 0$ , i.e. le plan tangent est bien défini.

La condition (C1) assure que les surfaces partagent bien une courbe commune selon le paramètre  $v$  (aussi appelé continuité géométrique d'ordre un). La condition (C2) garantit que les surfaces partagent un plan tangent commun en tous points de leur courbe frontière. La condition (C3) force les surfaces à se répartir de chaque côté de la courbe de bord, c'est-à-dire à garder une bonne orientation. La condition (C4) assure que le plan tangent est bien défini en empêchant les dérivées premières d'être colinéaires.

Dans le cas de facettes triangulaires de Bézier, seuls les points de contrôle de la courbe de bord commune et la première rangée de part et d'autre de celle-ci interviennent dans la continuité  $G^1$ . Ce sont donc ces trois rangées qu'il faut calculer pour construire deux facettes se raccordant avec continuité  $G^1$ . Deux approches ont principalement été utilisées pour cela : celle de Farin et celle de Chiyokura-Kimura.

### A.1.1 Méthode de Farin



**Fig. A.1.2** – Points de contrôle des patches  $Q$  et  $R$ .

La méthode de Farin consiste à considérer  $\mu$  et  $\nu$  constants et  $\varphi$  linéaire. A partir des points de contrôle aux extrémités supposés connus, des coefficients  $\alpha_i$  sont calculés par la formule suivante (notations tirées de la figure A.1.2,  $n$  degré des courbes) :

$$\begin{aligned} q_0 &= \alpha_1 c_0 + \alpha_2 c_1 + \alpha r_0, & \alpha_1 + \alpha_2 + \alpha &= 1 \\ q_{n-1} &= \alpha_3 c_{n-2} + \alpha_4 c_{n-1} + \alpha r_{n-1}, & \alpha_3 + \alpha_4 + \alpha &= 1 \end{aligned}$$

Les deux patches  $Q$  et  $R$  se raccordent avec continuité  $G^1$  si et seulement si

$$q_i = \frac{n-1-i}{n-1}(\alpha_1 c_i + \alpha_2 c_{i+1} + \alpha r_i) + \frac{i}{n-1}(\alpha_3 c_{i-1} + \alpha_4 c_i + \alpha r_i), \text{ pour } i = 0, \dots, n-1 \quad (\text{A.1.2})$$

La méthode de Farin permet de calculer les points de contrôle de Bézier autour d'une courbe de bord pour assurer la continuité  $G^1$ . Pour cela, des paramètres  $\alpha_i$  sont obtenus des conditions aux bords, les points de contrôle des rangées intérieures sont alors obtenus grâce à l'équation (A.1.2).

### A.1.2 Méthode de Chiyokura-Kimura

La méthode de Chiyokura-Kimura construit un champ de vecteurs dérivées transfrontières le long de la courbe de bord commune aux deux patches. Ce champ de vecteur définit un plan tangent le long de cette courbe. Chaque patch est alors calculé séparément de manière à avoir un plan tangent correspondant à ce champ de vecteurs. Ces patches se raccorderont donc bien de manière  $G^1$  bien qu'ils aient été construits individuellement. Initialement, cette approche a été utilisée sur des surfaces de degré trois, cas particulier développé ci-dessous. Cependant elle reste applicable au degré  $n$  quelconque. Avec les mêmes notations que ci-dessus, on suppose connus les points de contrôle  $c_i$  et  $q_0, r_0, q_{n-2}$  et  $r_{n-2}$ .

On définit  $v_0$  comme le vecteur unitaire perpendiculaire à  $c_1 - c_0$  dans le plan tangent commun à  $Q$  et  $R$  au point  $c_0$  (unique au signe près) et  $v_1$  par comme le vecteur unitaire perpendiculaire à  $c_3 - c_2$  dans le plan tangent commun à  $Q$  et  $R$  au point  $c_3$ . Soit  $V(t) = (1-t)v_0 + tv_1$  combinaison affine de  $v_0$  et  $v_1$ .

Les dérivées transfrontières de  $Q$  et  $R$  sont données par :

$$\begin{aligned} \frac{\partial Q}{\partial u}(0, v) &= k(v)V(v) + h(v)\frac{\partial Q}{\partial v}(0, v), \\ \frac{\partial R}{\partial w}(0, v) &= l(v)V(v) + m(v)\frac{\partial Q}{\partial v}(0, v), \end{aligned}$$

où  $k(v)$ ,  $h(v)$ ,  $l(v)$  et  $m(v)$  sont des fonctions linéaires. Elles sont calculés grâce à l'évaluation de ces équations en  $v = 0$  et  $v = 1$ . Chaque point de contrôle inconnu est alors facilement obtenu en écrivant le membre de droite dans la base de Brenstein.

# Annexe B

## Format de fichier hiérarchique

Le modelleur 3D permet de créer et modifier des objets en utilisant la représentation hiérarchique qui est développée dans ce mémoire. Il est important de pouvoir sauvegarder le fruit d'une session de design dans un fichier pour pouvoir le recharger plus tard en mémoire. De plus, vu la taille possible des données (par exemple des terrains de plusieurs centaines de milliers de faces), il est intéressant de permettre la transmission progressive de ces données. Dans ce but, un format de fichier a été spécialement créé. Il permet de sauvegarder tous les paramètres libres dont dépend la surface. Ainsi, en lisant ces paramètres libres, il est possible de reconstruire une surface à l'identique.

Pour mettre au point ce format de fichier, nous nous sommes librement inspirés des formats de geomview ainsi que du format neighbourhood développé par Georges-Pierre Bonneau. Ce format de fichier est constitué de deux parties. Tout d'abord, le maillage initial est donné sans aucun soucis de subdivision sous une forme qui permet de recréer la surface initiale : sont donc stockés la topologie et la géométrie du maillage plus les paramètres libres de ce niveau.

On commence par signifier que le fichier est bien de type hiérarchique en écrivant HIE au début du fichier. Ensuite, le nombre de sommets et le nombre de faces sont précisés. Vient alors la liste des sommets initiaux sous la forme : les coordonnées  $x$   $y$   $z$ , le degré  $No$  du sommet (son nombre de voisin) suivi des coordonnées des  $No$  dérivées premières puis des  $No$  twists. Tous les paramètres libres autour du sommet sont donc fixés. La liste des faces suit sous la forme : les trois indices des sommets de la face (indice donnés par l'ordre dans la liste précédente) puis les indices des faces voisines. Les faces voisines sont données dans le même ordre que les sommets, la première face voisine correspondant à l'arête située face au premier sommet et ainsi de suite. L'indice -1 est donné pour la face voisine si l'arête concernée est un

bord.

A ce niveau, une première surface lisse peut-être construite. Seuls manquent les six points libres intérieurs de chaque face. Ceux-ci sont temporairement initialisés à l'aide d'une minimisation d'énergie en attendant la suite de la lecture du fichier.

On trouve alors à la suite les paramètres libres attachés aux faces et les données sur les sommets subdivisés sous la forme : le numéro de la face de départ, le chemin à suivre pour arriver jusqu'à la face finale, le niveau de profondeur de la face. Ces données déterminent la face courante à traiter. Viennent alors les coordonnées des six points libres intérieurs suivies du nombre d'arêtes subdivisées dans cette face. Chaque sommet inséré est donné par : l'indice de l'arête à subdiviser, les coordonnées du sommet et les coordonnées de ses dérivées premières et twists. Il faut remarquer que l'ordre de ces sommets insérés est six, sauf s'ils se trouvent sur le bord, fait connu au moment de l'insertion.

La face initiale est indispensable de manière évidente : n'oublions pas que nous avons une forêt de faces.

Le chemin à suivre pour arriver à la face finale est constitué d'un entier (unsigned long int) écrit en base quatre. Chaque chiffre représente alors l'indice de la face fille dans laquelle continuer. Ceci dans l'ordre inverse. Par exemple 3120 en base 4 veut dire que la face est fille numéro trois de la fille numéro un de la fille numéro deux de la fille numéro zéro de la face initiale. Vient ensuite la profondeur de la face. Là encore, cela s'avère indispensable car il faut savoir à combien de chiffres en base quatre s'arrêter parmi les seize chiffres (toujours en base quatre) d'un unsigned long int.

Finalement, nous donnons toutes les coordonnées globales du sommet plutôt que les locales à cause d'un problème d'orientation du repère. En effet, l'un des axes est défini par une arête, i.e. un arc non orienté. Nous donnons une orientation artificiellement (par une relation d'ordre sur les adresses mémoires) mais cette orientation peut changer entre différentes exécutions du programme. Bien sûr, donner les coordonnées globales force à inverser la matrice de base, mais comme c'est une matrice orthonormale, cela revient à en prendre la transposée. De la même façon, l'ordre des dérivées premières et des twists est important. Il est directement déterminé par la mode de référencement des faces : on définit une relation d'ordre sur la faces grâce au numéro de la face initiale et du chemin. Les dérivées et twists sont données en commençant par la face voisine d'adresse inférieure. Cette relation d'ordre fournit aussi la face à laquelle est associée un sommet inséré. En effet, il est commun à deux faces. Dans le fichier, il n'apparaît que par la face d'adresse inférieure.

L'ordre des sommets subdivisés est important, il faut les donner dans l'ordre

croissant de profondeur. Cela permet de n'avoir jamais à générer des subdivisions qui n'existent pas lors de la synthèse du fichier. En effet, si l'on donne un chemin qui passe par des faces non encore créées, il faut les générer, mais on subdivise une arête c'est-à-dire qu'on crée 8 faces d'un coup. On risque alors d'en créer en trop. Ainsi, avec notre format de fichier, on peut faire de la transmission progressive par niveau mais pas par ordre décroissant des normes des coordonnées locales comme l'on aurait souhaité. L'ordre décroissant permettrait de prendre en compte d'abord les variations les plus importantes avant d'ajouter les détails. Le problème de la non-crédation de faces inutiles est complexe et traité dans [Gioia, 2000]. Une implémentation est possible, mais n'était pas dans les priorités de cette thèse.

Voici les spécifications du format d'échange :

----- SPECIFICATIONS DU FICHIER d'ECHANGE -----

Le suffixe est fixé par convention à '.hie'. La syntaxe du fichier est :

```

HIE                # Mot clef
NV NF              # NF Nombre de faces initiales
                   # NV Nombre de sommets initiaux

# liste des sommets initiaux
Sommet(v=0)
Sommet(v=1)
Sommet(v=2)
...
Sommet(v=Nv-1)

# liste des faces initiales
v[0] v[1] v[2] F[0] F[1] F[2] # les trois sommets suivis
...                          # des trois faces voisines

# liste des faces filles
Nparent Children level # pour acceder a la face
    point libre(p=0)    # coordonnees des six points libres
    point libre(p=1)
    ...
    point libre(p=6)

NA                    # Nombre d'aretes raffinees de cette face
Sommet d'arete(e=0)

```



```

...
Sommet d'arete(e=NA-1)
...

```

où :

*Sommet* est de la forme :

```

x y z
No # ordre du sommet
der_x(n=0) der_y(n=0) der_z(n=0)
...
der_x(n=No-1) der_y(n=No-1) der_z(n=No-1)
twist_x(n=0) twist_y(n=0) twist_z(n=0)
...
twist_x(n=No-1) twist_y(n=No-1) twist_z(n=No-1)

```

*Sommet d'arete* est de la forme :

```

Ne # indice de l'arete a subdiviser
x y z
der_x(n=0) der_y(n=0) der_z(n=0)
...
der_x(n=5) der_y(n=5) der_z(n=5)
twist_x(n=0) twist_y(n=0) twist_z(n=0)
...
twist_x(n=5) twist_y(n=5) twist_z(n=5)

```

*Nparent* est l'indice de la première face dans la liste des faces initiales,

*Children* est un unsigned long int qui représente le chemin à la face en base 4 : l'indice de la première face fille suivi de l'indice de la deuxième... par exemple (en base 4) 312 veut dire la troisième fille de la première fille de la seconde fille de la face *Nparent*.

# Annexe C

## Algorithme de subdivision de Goldman

Dans [Goldman, 1983], l'auteur présente un algorithme permettant de subdiviser des triangles de Bézier. Pour cela, on insère un nombre  $n$  de points puis on calcule un nouvel objet en dimension  $n+2$  contenu dans l'enveloppe convexe de nos  $n+3$  points (les trois sommets du triangle de départ). On obtient un  $(n+3)$ D Bézier simplexe dégénéré dont les faces nous donnent une subdivision du triangle de départ. La plupart des résultats obtenus sont démontrés à l'aide d'une approche probabiliste.

Soit  $S = (S_0, \dots, S_m)$  l'ensemble des résultats possibles pour un événement  $E$  et soit  $T = (t_0, \dots, t_m)$  leurs probabilités respectives. Par définition, on a :

$$t_k \geq 0, k = 0, \dots, m \quad \text{et} \quad |T| = \sum t_k = 1$$

Soit  $I = (i_0, \dots, i_m)$  un  $(m+1)$  uple d'entiers tels que :

$$i_k \geq 0, k = 0, \dots, m \quad \text{et} \quad |I| = \sum i_k = 1$$

On définit  $B_I^n(T)$  = la probabilité d'avoir  $i_k$  succès de  $S_k$ ,  $k = 0, \dots, m$  en  $n$  expériences aléatoires.

La loi définie par  $B_I^n(T)$  s'appelle la loi multinomiale. Quand  $m = 1$ , on retombe sur la loi binomiale. Comme on travaille avec des probabilités, on a :

$$B_I^n(T) \geq 0$$
$$\sum_{|I|=n} B_I^n(T) = 1$$

Les fonctions  $B_I^n(T)$  sont données explicitement par les polynômes

$$B_I^n(T) = \binom{n}{I} T^I$$

On définit

$$\Delta^m = \{T = (t_0, \dots, t_m) | t_k \geq 0 \text{ et } |T| = 1\}$$

le simplexe standard de dimension  $m$  et on note

$$\{P_i = P_{i_0, \dots, i_m} | i_k \geq 0 \text{ et } |I| = n\}$$

un ensemble de points.

On définit le simplexe de Bézier de degré  $n$  et de dimension  $m$  généré par les points  $P_i$  comme étant

$$\left\{ \sum_{|I|=n} B_I^n(T) P_I | T \in \Delta^m \right\}$$

On note l'application canonique du simplexe standard au simplexe de Bézier

$$B^n[P_I](T) = \sum_{|I|=n} B_I^n(T) P_I$$

Le simplexe de Bézier est contenu dans l'enveloppe convexe de ses générateurs.

Un sous-ensemble de  $\Delta^m$  défini par les équations

$$t_{j_1} = t_{j_2} = \dots = t_{j_k}$$

forme un simplexe de dimension  $n - k$  et est généré par l'ensemble

$$\{P_I | i_{j_1} = i_{j_2} = \dots = i_{j_k}\}$$

Pour subdiviser un triangle de Bézier  $B^n[P_I](T)$ , on doit trouver un ensemble de points  $Q_I$  tels que

$$B^n[Q_I](T) \subset B^n[P_I](T)$$

Un simplexe de Bézier dégénéré génère automatiquement une subdivision des triangles de Bézier. Supposons que le simplexe de Bézier soit identique à l'un de ses sous-simplexes triangulaires. Grâce à la propriété que nous venons d'énoncer, chaque sous-triangle d'un simplexe de Bézier est une surface de Bézier triangulaire. Dans la suite, on note  $E_k$  le vecteur à  $(m + 1)$  composantes toutes nulles sauf la  $k^{\text{ième}}$  qui vaut 1.

Nous voulons insérer trois points, un au milieu de chaque arête du simplexe standard. Cela revient à construire un simplexe de Bézier dégénéré en dimension cinq.

Soit un ensemble de points  $\{P_I\}$ ,  $I = (i_0, i_1, i_2)$ ,  $|I| = n$  générant le triangle de Bézier

$$B(T) = B^n[P_I](T) = \sum B_I^n(T) P_I$$

et

$$Q_0 = (u_0, v_0, w_0) \in \Delta^2$$

$$Q_1 = (u_1, v_1, w_1) \in \Delta^2$$

$$Q_2 = (u_2, v_2, w_2) \in \Delta^2$$

trois points de l'espace des paramètres. On peut définir un ensemble 5D de points  $\{P_J^K(Q_0, Q_1, Q_2), J = (j_0, j_1, j_2), K = (k_0, k_1, k_2), |J| + |K| = n\}$  récursivement par

$$\begin{aligned} P_J^{000}(Q_0, Q_1, Q_2) &= P_J \\ P_J^{K+E_0}(Q_0, Q_1, Q_2) &= u_0 P_{J+E_0}^K(Q_0, Q_1, Q_2) + v_0 P_{J+E_1}^K(Q_0, Q_1, Q_2) \\ &\quad + w_0 P_{J+E_2}^K(Q_0, Q_1, Q_2) \\ P_J^{K+E_1}(Q_0, Q_1, Q_2) &= u_1 P_{J+E_0}^K(Q_0, Q_1, Q_2) + v_1 P_{J+E_1}^K(Q_0, Q_1, Q_2) \\ &\quad + w_1 P_{J+E_2}^K(Q_0, Q_1, Q_2) \\ P_J^{K+E_2}(Q_0, Q_1, Q_2) &= u_2 P_{J+E_0}^K(Q_0, Q_1, Q_2) + v_2 P_{J+E_1}^K(Q_0, Q_1, Q_2) \\ &\quad + w_2 P_{J+E_2}^K(Q_0, Q_1, Q_2) \end{aligned}$$

Grâce à l'équation suivante :

$$B^n[P_J(Q_0, Q_1, Q_2)](T) = B^n[P_I][(t_0, t_1, t_2) + t_3 Q_0 + t_4 Q_1 + t_5 Q_2]$$

On montre que le simplexe Bézier engendré par  $\{P_J^K(Q_0, Q_1, Q_2)\}$  donne la même chose que le triangle de Bézier obtenu par  $\{P_I\}$ . Ses sous-simplexes triangulaires subdivisent donc notre triangle de Bézier. Ces simplexes sont définis par les équations

$$t_{h_1} = t_{h_2} = t_{h_3} = 0, h_1 \neq h_2 \neq h_3$$

On obtient en tout 20 triangles de Bézier. Néanmoins, pour notre problème, cela se simplifie. En effet, on prend

$$Q_0 = (0, \frac{1}{2}, \frac{1}{2})$$

$$Q_1 = (\frac{1}{2}, 0, \frac{1}{2})$$

$$Q_2 = (\frac{1}{2}, \frac{1}{2}, 0)$$

Les trois points étant sur le bord, de nombreux triangles vont être confondus avec le bord. A la fin, il ne reste plus que les quatre triangles de Bézier qui nous intéressent.



# Bibliographie

- [Alfeld, 1989] ALFELD, P. (1989). Scattered data interpolation in three or more variables. *In Mathematical methods in computer aided geometric design*, pages 1–33. Academic Press Professional, Inc.
- [Alkalai et Dyn, 2004] ALKALAI, N. et DYN, N. (2004). Optimising 3d triangulations : improving the initial triangulation for the butterfly subdivision scheme. *In DODSON, N., SABIN, M. et FLOATER, M., éditeurs : Advances in Multiresolution for Geometric Modeling*. Springer-Verlag.
- [Alliez et al., 2002] ALLIEZ, P., MEYER, M. et DESBRUN, M. (2002). Interactive geometry remeshing. *ACM Transactions on Graphics. Special issue for SIGGRAPH conference*, 21(3):347–354.
- [Bennis et al., 1991] BENNIS, C., VÉZIEN, J.-M. et IGLÉSIAS, G. (1991). Piecewise surface flattening for non-distorted texture mapping. *SIGGRAPH Comput. Graph.*, 25(4):237–246.
- [Bonneau, 1998] BONNEAU, G.-P. (1998). Multiresolution analysis on irregular surface meshes. *IEEE Transactions on Visualization and Computer Graphics*, 4(4):365–378.
- [Bonneau et Hahmann, 2000] BONNEAU, G.-P. et HAHMANN, S. (2000). Polyhedral modeling. *In Proceedings of the conference on Visualization '00*, pages 381–387. IEEE Computer Society Press.
- [Bonneau et Hahmann, 2003] BONNEAU, G.-P. et HAHMANN, S. (2003). Smooth polylines on polygon meshes. *In BRUNETT, G., HAMANN, B., MÜLLER, H. et LINSEN, L., éditeurs : Geometric Modeling for Scientific Visualization*, Mathematics and Visualization, pages 69–84. Springer.
- [Certain et al., 1996] CERTAIN, A., POPOVIC, J., DEROSE, T., DUCHAMP, T., SALESIN, D. et STUETZLE, W. (1996). Interactive multiresolution surface viewing. *Computer Graphics*, 30(Annual Conference Series - SIGGRAPH):91–98.

- [Chen et Han, 1990] CHEN, J. et HAN, Y. (1990). Shortest paths on a polyhedron. *In Proceedings of the sixth annual symposium on Computational geometry*, pages 360–369. ACM Press.
- [Chiyokura et Kimura, 1983] CHIYOKURA, H. et KIMURA, F. (1983). Design of solids with free-form surfaces. *SIGGRAPH Comput. Graph.*, 17(3):289–298.
- [Curless et Levoy, 1996] CURLESS, B. et LEVOY, M. (1996). A volumetric method for building complex models from range images. *Computer Graphics (SIGGRAPH '96 Proceedings)*, pages 303–312.
- [de Floriani et Puppo, 1995] de FLORIANI, L. et PUPPO, E. (1995). Hierarchical triangulation for multiresolution surface description. *ACM Transactions on Graphics*, 14(4):363–411.
- [DeRose et al., 1998] DEROSE, T., KASS, M. et TRUONG, T. (1998). Subdivision surfaces in character animation. *In Proceedings of the 25th annual conference on Computer graphics and interactive techniques (SIGGRAPH Proceedings)*, pages 85–94. ACM Press.
- [DeRose, 1990] DEROSE, T. D. (1990). Necessary and sufficient conditions for tangent plane continuity of bézier surfaces. *Comput. Aided Geom. Des.*, 7(1-4):165–179.
- [Desbrun et al., 2002] DESBRUN, M., MEYER, M. et ALLIEZ, P. (2002). Intrinsic parameterizations of surface meshes. *Computer Graphics Forum*, 21:209–218.
- [Desbrun et al., 1999] DESBRUN, M., SCHRÖDER, P. et BARR, A. (1999). Interactive animation of structured deformable objects. *In Graphics Interface*, pages 1–8.
- [Dijkstra, 1959] DIJKSTRA, E. (1959). A note on two problems in connection with graphs. *Numer. Math 1*, pages 269–271.
- [Dyn et al., 1990] DYN, N., LEVINE, D. et GREGORY, J. A. (1990). A butterfly subdivision scheme for surface interpolation with tension control. *ACM Trans. Graph.*, 9(2):160–169.
- [Eck et al., 1995] ECK, M., DEROSE, T., DUCHAMP, T., HOPPE, H., LOUNSBERY, M. et STUETZLE, W. (1995). Multiresolution analysis of arbitrary meshes. *Computer Graphics*, 29(Annual Conference Series (SIGGRAPH Proceedings)):173–182.
- [Eck et Hoppe, 1996] ECK, M. et HOPPE, H. (1996). Automatic reconstruction of b-spline surfaces of arbitrary topological type. *In Proceedings of the 23rd annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, pages 325–334. ACM Press.

- [Farin, 1996] FARIN, G. E. (1996). *Curves and Surfaces for Computer-Aided Geometric Design : A Practical Code*. Academic Press, Inc.
- [Ferley, 2002] FERLEY, E. (2002). *Sculpture Virtuelle*. Thèse de doctorat, INPG.
- [Finkelstein et Salesin, 1994] FINKELSTEIN, A. et SALESIN, D. H. (1994). Multiresolution curves. *In Proceedings of the 21st annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, pages 261–268. ACM Press.
- [Floater, 1997] FLOATER, M. S. (1997). Parametrization and smooth approximation of surface triangulations. *Comput. Aided Geom. Des.*, 14(3):231–250.
- [Forsey et Wong, 1998] FORSEY, D. et WONG, D. (1998). Multiresolution surface reconstruction for hierarchical B-splines. *In Graphics Interface*, pages 57–64.
- [Forsey et Bartels, 1988] FORSEY, D. R. et BARTELS, R. H. (1988). Hierarchical b-spline refinement. *In Proceedings of the 15th annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, pages 205–212. ACM Press.
- [Forsey et Bartels, 1995] FORSEY, D. R. et BARTELS, R. H. (1995). Surface fitting with hierarchical splines. *ACM Transactions on Graphics*, 14(2):134–161.
- [Fowler et Bartels, 1993] FOWLER, B. et BARTELS, R. (1993). Constraint-based curve manipulation. *IEEE Comput. Graph. Appl.*, 13(5):43–49.
- [Franke, 1982] FRANKE, R. (1982). Scattered data interpolation : Tests of some methods. *Mathematics of Computation*, 38:181–200.
- [Franke et Nielson, 1990] FRANKE, R. et NIELSON, G. (1990). Scattered data interpolation and applications : A tutorial and survey. *Geometric Modeling, Survey on SD Methods*, pages 131–159.
- [Garland, 1998] GARLAND, M. (1998). *Quadric-Based Polygonal Surface Simplification*. Thèse de doctorat, Carnegie Mellon University.
- [Gioia, 2000] GIOIA, P. (2000). Modélisation, codage et visualisation de surfaces maillées par ondelettes pour la représentation multirésolution et la transmission progressive. Phd thesis, Université de Rennes.
- [Goldman, 1983] GOLDMAN, R. N. (1983). Subdivision algorithms for Bézier triangles. *Computer-Aided Design*, 22(3):159–166.
- [Gonzalez-Ochoa et Peters, 1999] GONZALEZ-OCHOA, C. et PETERS, J. (1999). Localized-hierarchy surface splines (less). *In Proceedings of the 1999 symposium on Interactive 3D graphics*, pages 7–15. ACM Press.
- [Gortler et Cohen, 1995] GORTLER, S. J. et COHEN, M. F. (1995). Hierarchical and variational geometric modeling with wavelets. *In Proceedings of the 1995 symposium on Interactive 3D graphics*, pages 35–42. ACM Press.



- [Gregory, 1986] GREGORY, J. A. (1986). N-sided surface patches. In GREGORY, J. A., éditeur : *The Mathematics of Surfaces*, pages 217–232. Clarendon Press, Oxford.
- [Guilbert, 2000] GUILBERT, E. (2000). Interpolation  $g^1$  hiérarchique. Mémoire de dea, Université Joseph Fourier.
- [Hahmann et Bonneau, 2000] HAHMANN, S. et BONNEAU, G.-P. (2000). Triangular  $g^1$  interpolation by 4-splitting domain triangles. *Computer Aided Geometric Design*, 17(8):731–757.
- [Hahmann et Bonneau, 2003] HAHMANN, S. et BONNEAU, G.-P. (2003). Polynomial surfaces interpolating arbitrary triangulations. *IEEE Transactions on Visualization and Computer Graphics*, 9(1):99–109.
- [Hahmann et al., 2003] HAHMANN, S., BONNEAU, G.-P. et YVART, A. (2003). Subdivision invariant polynomial interpolation. In H.C. HEGE, K. P. e., éditeur : *Visualization and Mathematics III*, Mathematics and Visualization, pages 191–202. Springer.
- [Heckbert et Garland, 1994] HECKBERT, P. et GARLAND, M. (1994). Multiresolution modeling for fast rendering. In *Proceedings of Graphics Interface '94*, pages 43–50.
- [Hoppe, 1996] HOPPE, H. (1996). Progressive meshes. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, pages 99–108. ACM Press.
- [Hoppe, 1997] HOPPE, H. (1997). View-dependent refinement of progressive meshes. *Computer Graphics*, 31(Annual Conference Series (SIGGRAPH Proceedings)): 189–198.
- [Hoppe et al., 1994] HOPPE, H., DEROSE, T., DUCHAMP, T., HALSTEAD, M., JIN, H., McDONALD, J., SCHWEITZER, J. et STUETZLE, W. (1994). Piecewise smooth surface reconstruction. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, pages 295–302. ACM Press.
- [Hoppe et al., 1993] HOPPE, H., DEROSE, T., DUCHAMP, T., McDONALD, J. et STUETZLE, W. (1993). Mesh optimization. In *Computer Graphics (SIGGRAPH '93 Proceedings)*, pages 19–26.
- [Hoschek et al., 1993] HOSCHEK, J., LASSER, D. et SCHUMAKER, L. L. (1993). *Fundamentals of computer aided geometric design*. A. K. Peters, Ltd.
- [Hubeli et Gross, 2000] HUBELI, A. et GROSS, M. (2000). A survey of surface representations for geometric modeling.

- [Jensen, 1987] JENSEN, T. (1987). Assembling triangular and rectangular patches and multivariate splines. *In FARIN, G., éditeur : Geometric Modeling : Algorithms and New Trends*, pages 203–220. SIAM.
- [Khodakovsky et Schröder, 1999] KHODAKOVSKY, A. et SCHRÖDER, P. (1999). Fine level feature editing for subdivision surfaces. *In Proceedings of the fifth ACM symposium on Solid modeling and applications*, pages 203–211. ACM Press.
- [Kobbelt, 1996] KOBBELT, L. (1996). A variational approach to subdivision. *Computer Aided Geometric Design*, pages 743–761.
- [Kobbelt, 2000] KOBBELT, L. (2000).  $\sqrt{3}$ -subdivision. *In Proceedings of the 27th annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, pages 103–112. ACM Press/Addison-Wesley Publishing Co.
- [Kobbelt et al., 1998] KOBBELT, L., CAMPAGNA, S., VORSATZ, J. et SEIDEL, H.-P. (1998). Interactive multi-resolution modeling on arbitrary meshes. *In Proceedings of the 25th annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, pages 105–114. ACM Press.
- [Krishnamurthy et Levoy, 1996] KRISHNAMURTHY, V. et LEVOY, M. (1996). Fitting smooth surfaces to dense polygon meshes. *Computer Graphics*, 30(Annual Conference Series (SIGGRAPH Proceedings)):313–324.
- [Lasseter, 1987] LASSETER, J. (1987). Principles of traditional animation applied to 3d computer animation. *In Proceedings of the 14th annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, pages 35–44. ACM Press.
- [Lee et al., 1998] LEE, A. W. F., SWELDENS, W., SCHRÖDER, P., COWSAR, L. et DOBKIN, D. (1998). Maps : Multiresolution adaptive parameterization of surfaces. *Computer Graphics Proceedings (SIGGRAPH 98)*, pages 95–104.
- [Litke et al., 2001] LITKE, N., LEVIN, A. et SCHROEDER, P. (2001). Fitting subdivision surfaces. *In IEEE Visualization 2001*, pages 319–324.
- [Loop, 1994] LOOP, C. (1994). A  $G^1$  triangular spline surface of arbitrary topological type. *Computer Aided Geometric Design*, 11:303–330.
- [Loop et DeRose, 1989] LOOP, C. T. et DEROSE, T. D. (1989). A multisided generalization of bézier surfaces. *ACM Trans. Graph.*, 8(3):204–234.
- [Lounsbery et al., 1997] LOUNSBERY, M., DEROSE, T. D. et WARREN, J. (1997). Multiresolution analysis for surfaces of arbitrary topological type. *ACM Transactions on Graphics*, 16(1):34–73.

- [Maillot *et al.*, 1993] MAILLOT, J., YAHIA, H. et VERROUST, A. (1993). Interactive texture mapping. *In Proceedings of the 20th annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, pages 27–34. ACM Press.
- [Mallat, 1989] MALLAT, S. (1989). A theory for multiresolution signal decomposition : The wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):674–693.
- [Mallat et Hwang, 1992] MALLAT, S. et HWANG, W. L. (1992). Singularity detection and processing with wavelets. *IEEE Transactions on Information Theory*, 38(2):617–643.
- [Mann *et al.*, 1992] MANN, S., LOOP, C., LOUNSBERY, M., MEYERS, D., J. PAINTER AND, T. D. et SLOAN, K. (1992). A survey of parametric scattered data fitting using triangular interpolants. *In HAGEN, H., éditeur : Curve and Surface Design*, pages 145–172. SIAM.
- [Nielson, 1980] NIELSON, G. (1980). Minimum norm interpolation in triangles. *SIAM J. Numer. Anal.*, 17(1):44–62.
- [Nielson, 1983] NIELSON, G. (1983). A method for interpolating scattered data based upon a minimum norm network. *Mathematics of Computation*, 40:253–271.
- [Peters, 1990a] PETERS, J. (1990a). *Fitting smooth parametric surfaces to 3D data*. Thèse de doctorat, University of Wisconsin. PhD thesis ; see also CMS Technical Report 91-2.
- [Peters, 1990b] PETERS, J. (1990b). Local smooth surface interpolation : a classification. *Comput. Aided Geom. Des.*, 7(1-4):191–195.
- [Peters, 1991] PETERS, J. (1991). Smooth interpolation of a mesh of curves. *Constructive Approximation*, 7:221–246.
- [Peters, 2002a] PETERS, J. (2002a).  $C^2$  free-form surfaces of degree (3,5). *Computer-Aided Geometric Design*, 19(2):113–126.
- [Peters, 2002b] PETERS, J. (2002b). Geometric continuity. *In Handbook of Computer Aided Geometric Design*, pages 193–229. Elsevier.
- [Piper, 1987] PIPER, B. (1987). Visually smooth interpolation with triangular bézier patches. *In FARIN, G., éditeur : Geometric Modeling : Algorithms and new Trends*, pages 221–233. SIAM.
- [Rossignac et Borrel, 1993] ROSSIGNAC, J. et BORREL, P. (1993). *Multi-resolution 3D approximations for rendering complex scenes*, pages 455–465.

- [Sabin et Barthe, 2003] SABIN, M. et BARTHE, L. (2003). Artifacts in recursive subdivision surfaces. *In* COHEN, A., MERRIEN, J.-L. et (EDS.), L. S., éditeurs : *Proceedings of the fifth conference on curves and surfaces*, pages 353–362.
- [Sarraga, 1987] SARRAGA, R. F. (1987). G1 interpolation of generally unrestricted cubic bézier curves. *Comput. Aided Geom. Des.*, 4(1-2):23–39.
- [Schröder et Zorin, 1998] SCHRÖDER, P. et ZORIN, D. (1998). Subdivision for modeling and animation. Course notes of Siggraph 98. ACM SIGGRAPH.
- [Schweitzer, 1996] SCHWEITZER, J. E. (1996). Analysis and application of subdivision surfaces. Rapport technique TR-96-08-02.
- [Sederberg *et al.*, 2003] SEDERBERG, T. W., ZHENG, J., BAKENOV, A. et NASRI, A. (2003). T-splines and t-nurccs. *ACM Trans. Graph.*, 22(3):477–484.
- [Shirman et Sequin, 1987] SHIRMAN, L. A. et SEQUIN, C. H. (1987). Local surface interpolation with bézier patches. *Computer Aided Geometric Design*, 4:279–295.
- [Shirman et Sequin, 1991] SHIRMAN, L. A. et SEQUIN, C. H. (1991). Local surface interpolation with bézier patches : errata and improvements. *Computer Aided Geometric Design*, 8:217–221.
- [Stam, 1998] STAM, J. (1998). Exact evaluation of Catmull-Clark subdivision surfaces at arbitrary parameter values. *Computer Graphics*, 32(Annual Conference Series (SIGGRAPH Proceedings)):395–404.
- [Taubin, 1995] TAUBIN, G. (1995). A signal processing approach to fair surface design. *In Proceedings of the 22nd annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, pages 351–358. ACM Press.
- [Turk et Levoy, 1994] TURK, G. et LEVOY, M. (1994). Zippered polygon meshes from range images. *Computer Graphics (SIGGRAPH '94 Proceedings)*, pages 311–318.
- [Valette et Prost, 2004] VALETTE, S. et PROST, R. (2004). Wavelet-based multiresolution analysis of irregular surface meshes. *IEEE Transactions on Visualization and Computer Graphics*, 10(2):113–122.
- [Watkins, 1988] WATKINS, M. A. (1988). Problems in geometric continuity. *Computer Aided Design*, 20(8):499–502.
- [Weiss *et al.*, 2002] WEISS, V., ANDOR, L., RENNERT, G. et V&#225 ;RADY, T. (2002). Advanced surface fitting techniques. *Comput. Aided Geom. Des.*, 19(1):19–42.
- [Wijk, 1986] WIJK, J. J. V. (1986). Bicubic patches for approximating non-rectangular control meshes. *Constructive Approximation*, 3:1–13.

- [Yvart *et al.*, 2004a] YVART, A., HAHMANN, S. et BONNEAU, G.-P. (2004a). Hierarchical triangular splines. *submitted*.
- [Yvart *et al.*, 2004b] YVART, A., HAHMANN, S. et BONNEAU, G.-P. (2004b). Smooth adaptive fitting of 3d models using hierarchical triangular splines. *submitted*.
- [Zorin *et al.*, 1996] ZORIN, D., SCHRÖDER, P. et SWELDENS, W. (1996). Interpolating subdivision for meshes with arbitrary topology. *Computer Graphics*, 30(Annual Conference Series (SIGGRAPH Proceedings)):189–192.
- [Zorin *et al.*, 1997] ZORIN, D., SCHRÖDER, P. et SWELDENS, W. (1997). Interactive multiresolution mesh editing. *In Proceedings of the 24th annual conference on Computer graphics and interactive techniques (SIGGRAPH)*, pages 259–268. ACM Press/Addison-Wesley Publishing Co.

## Modélisation Hiérarchique de Surfaces à partir de Maillages Polyédriques et Applications.

La modélisation géométrique est incontournable dans la création et l'élaboration d'un produit. De même, l'infographie est couramment employée en effets spéciaux ou réalité virtuelle. Pourtant, chaque domaine utilise ses propres outils pour modéliser des surfaces.

Pour répondre aux contraintes de ces deux types d'utilisation, nous proposons une nouvelle modélisation de surfaces polynomiales, interpolantes, globalement lisses (de plan tangent continu) et hiérarchique. Une surface lisse initiale est d'abord construite sur un maillage triangulaire, ce qui permet de traiter toutes les topologies. Des détails sont ensuite progressivement ajoutés par niveaux. Pour ce faire, chaque face est subdivisée en quatre sous-faces par insertion d'un sommet en milieu d'arête. Ce raffinement n'induit pas de modifications significatives sur la surface. Chaque détail est défini localement par rapport au précédent grâce à l'utilisation de repères locaux. Cette hiérarchie permet à l'ensemble des détails fins de suivre continûment le déplacement lors d'une édition de la surface.

Deux applications sont ensuite proposées dans cet ouvrage : Tout d'abord, un modéleur 3D respectant la démarche créative des artistes. Celui-ci repose sur le calcul d'une forme globale progressivement affinée pour obtenir l'objet désiré, et offre la possibilité d'éditer les objets de manière intuitive en modifiant très peu de paramètres. Enfin, un outil de reconstruction est présenté pour modéliser des objets existants grâce à notre nouvelle représentation hiérarchique.

**Mots-clés :** Surface hiérarchique, niveaux de détail, interpolation polynomiale, modélisation géométrique, patch de Bézier, subdivision, raffinement.