

THÈSE

présentée devant

L'UNIVERSITÉ CLAUDE BERNARD – LYON 1

pour l'obtention du

DIPLÔME DE DOCTORAT

(arrêté du 25 avril 2002)

SPÉCIALITÉ : INFORMATIQUE

présentée et soutenue publiquement le

5 décembre 2006

par

Rémi ALLÈGRE

**Contributions à l'introduction de flexibilité
dans la reconstruction et l'édition de modèles 3D**

Directeur de thèse :

Samir Akkouche

Composition du jury :

Christophe Schlick ...	Professeur, Université Bordeaux 2	Rapporteur
Mariette Yvinec	Chargée de recherche, INRIA Sophia-Antipolis	Rapporteur
Marie-Paule Cani	Professeure, INP Grenoble	Examineur
Bruno Lévy	Chargé de recherche, LORIA/INRIA Lorraine	Examineur
Samir Akkouche	Professeur, Université Lyon 1	Directeur de thèse
Raphaëlle Chaine	Maître de Conférences, Université Lyon 1	Co-encadrante

Remerciements

Je tiens tout d'abord à exprimer mes plus vifs remerciements à Samir Akkouche et à Raphaëlle Chaine pour la qualité de leur encadrement durant ces trois années. Ce travail n'aurait probablement pas été aussi abouti sans les nombreuses discussions que nous avons eues ensemble.

Je remercie Christophe Schlick et Mariette Yvinec pour avoir accepté d'être les rapporteurs de ma thèse, ainsi que Marie-Paule Cani et Bruno Lévy qui ont accepté de participer au jury.

J'adresse également mes remerciements aux thésards de l'axe 3 et aux différents membres de l'équipe de modélisation géométrique du LIRIS pour les discussions et les bons moments passés ensemble.

Je remercie enfin ma famille pour son soutien de tous les instants.

Ce travail a été financé par le Ministère de l'Éducation Nationale, de l'Enseignement Supérieur et de la Recherche dans le cadre du projet Art3D labellisé par l'ACI Masses de Données 2003-2006.

Table des matières

Introduction	1
1 Création de modèles 3D à partir d'objets physiques	2
2 Reconstruction dynamique de surface	5
3 Modélisation multireprésentation	7
I Un cadre de travail dynamique pour la reconstruction de surface à partir d'échantillons de points non structurés	9
1 État de l'art	11
1.1 Reconstruction de surface	11
1.1.1 Approche combinatoire	11
1.1.1.1 Méthodes volumiques	12
1.1.1.2 Méthodes de reconstruction par expansion de surface	14
1.1.1.3 Méthodes ayant recours au pôles	15
1.1.2 Ajustement de modèles	18
1.1.2.1 Reconstruction par ajustement d'un modèle de surface implicite	19
1.1.2.2 Modèles déformables	25
1.2 Simplification d'échantillons de points	26
1.2.1 Sélection ou calcul de points représentants de sous-ensembles	27
1.2.2 Simplification itérative	29
1.2.3 Simplification par échantillonnage incrémental	30
1.3 Synthèse	32
2 Reconstruction de surface par convection géométrique	35
2.1 Schéma de convection et interprétation géométrique	35
2.2 Analyse de l'algorithme : requêtes et prédicats mis en œuvre	37
2.3 Propriétés utiles en vue d'une approche dynamique	40

3	Mesure d'épaisseur locale	43
3.1	Cadre continu	43
3.2	Cadre discret	48
3.3	Conditions d'échantillonnage	48
3.4	Évaluation	50
4	Reconstruction sélective	53
4.1	Algorithme de reconstruction sélective	54
4.1.1	Procédure de simplification locale	54
4.1.2	Algorithme de convection géométrique sélective	54
4.1.3	Détection des poches	56
4.2	Contrôle de la simplification	57
4.3	Implantation efficace	60
4.3.1	Position du problème	60
4.3.2	Structure de données	61
4.3.3	Algorithme de reconstruction accéléré	61
4.3.4	Recherche spatiale accélérée	63
4.4	Résultats expérimentaux	64
4.4.1	Données testées et spécificités de leur traitement	64
4.4.2	Qualité des maillages	68
4.4.3	Performances	70
5	Mise à jour dynamique	73
5.1	Algorithme de mise à jour locale	73
5.2	Application à la correction interactive	77
5.2.1	Algorithme principal	77
5.2.2	Corrections locales multiples	79
5.2.3	Interface de correction interactive	79
5.3	Résultats expérimentaux	81
6	Reconstruction sélective par subdivision-fusion	85
6.1	Description de l'algorithme	85
6.2	Résultats expérimentaux	88
7	Conclusions et perspectives	93
	Annexes de la première partie	95

A	Triangulation de Delaunay	
A.1	Triangulations de Delaunay	97
A.1.1	Diagramme de Voronoï	97
A.1.2	Triangulation de Delaunay	97
A.1.3	Pôles	98
A.2	Axe médian et ε -échantillon	99
A.2.1	Axe médian	99
A.2.2	Local feature size	99
A.2.3	Conditions d'échantillonnage	100
A.3	Triangulation de Delaunay restreinte à une surface	100
B	Estimation locale de la direction des vecteurs normaux	103
II	Modélisation géométrique multireprésentation	105
1	Surfaces implicites et maillages polygonaux	107
1.1	Surfaces implicites	107
1.1.1	Définition et propriétés	107
1.1.2	Primitives implicites	108
1.1.3	Opérations booléennes et mélange	109
1.1.4	Déformations	110
1.1.5	Modèle des surfaces implicites à squelettes	110
1.1.6	Représentations discrètes	112
1.2	Maillages polygonaux	112
1.2.1	Définition et propriétés	112
1.2.2	Déformations	113
1.2.3	Opérations booléennes et mélange	114
1.2.4	Représentations différentielles	115
2	Méthodes multireprésentation	117
2.1	Techniques de conversion	117
2.1.1	Conversion de maillages en surfaces implicites	117
2.1.2	Conversion de surfaces implicites en maillages	118
2.2	Méthodes hybrides	120
2.3	Synthèse	122

3	L'HybridTree : un arbre de construction hybride	125
3.1	Architecture du modèle	125
3.2	Requêtes fondamentales	127
3.2.1	Primitives implicites à squelette	127
3.2.2	Maillages polygonaux	129
3.2.3	Opérations de mélange	133
3.2.4	Opérations booléennes	135
3.2.5	Opérations de déformation	136
3.2.5.1	Déformations de Barr	136
3.2.5.2	Déformations libres	137
3.2.5.3	Transformations affines	138
3.3	Maillage de l'HybridTree	139
3.3.1	Primitives implicites à squelette	140
3.3.2	Opérations de mélange	141
3.3.2.1	Mélange global	141
3.3.2.2	Maillage des nœuds de mélange négatif	147
3.3.2.3	Maillage des nœuds de mélange local	147
3.3.3	Opérations booléennes	147
3.3.4	Opérations de déformation	148
3.4	Détails d'implantation	148
3.5	Résultats et discussion	149
3.5.1	Modélisation libre	149
3.5.2	Restauration virtuelle	152
3.5.3	Discussion	152
4	Conclusions et perspectives	155
<hr/>		
	Bibliographie	157
	Publications de l'auteur	169

Introduction

Depuis une dizaine d'années, les progrès des technologies d'acquisition 3D ont favorisé l'essor de la modélisation géométrique d'objets réels à partir de données issues de leur numérisation. La capacité à créer, visualiser et manipuler des représentations numériques de la forme et de l'apparence d'objets physiques joue un rôle important dans de multiples domaines d'application, comme le design industriel, l'imagerie médicale, ou les systèmes d'information géographique. Un autre domaine d'application important de la numérisation 3D concerne l'étude et la conservation du patrimoine historique et culturel, comprenant les collections des musées, les objets et les sites archéologiques.

Cette thèse s'inscrit dans le cadre du projet Art3D labellisé par l'Action Concertée Incitative Masses de données 2003-2006, dont les partenaires sont le LIRMM (Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier), le C2RMF (Centre de Recherche et de Restauration des Musées de France), le LE2I (Laboratoire Électronique, Informatique et Image, Université de Bourgogne) et le LIRIS. Ce projet vise à l'étude d'une plateforme pour la numérisation et l'archivage des œuvres d'art du patrimoine historique et culturel. De par leur très grande précision (inférieure au millimètre), les dispositifs d'acquisition comme les scanners laser peuvent permettre de représenter des œuvres comme des sculptures de manière détaillée. Les modèles géométriques 3D en haute résolution constituent un support privilégié pour l'analyse de la forme et des détails des objets, tout en contribuant à la diffusion de la connaissance par la mise à disposition des modèles dans des bases de données ou des musées virtuels accessibles par internet. L'élaboration de modèles adaptés à ces différentes utilisations est une tâche complexe, qui nécessite de développer des outils et des algorithmes efficaces pour manipuler les données géométriques.

Plusieurs projets de numérisation d'œuvres d'art de grande ampleur ont été menés ces dernières années, dont le populaire *Digital Michelangelo Project* [127]. Ces projets ont contribué à la mise en place de chaînes de traitement permettant de passer de données comportant des millions, voire des milliards de points à des maillages manipulables sur des machines standard. Cependant, ces chaînes de production de modèles 3D sont particulièrement lourdes à mettre en œuvre. En premier lieu, la taille très importante des données d'entrée est un facteur qui complique la construction de modèles de surface, parfois excessivement du fait de la redondance des données ou d'un niveau de détail trop élevé par rapport aux applications visées. Il est nécessaire de développer des techniques capables d'analyser la pertinence des données pour produire des modèles compacts suffisamment précis, et pouvant être visualisés facilement. En outre, il est important de pouvoir modifier facilement la forme de ces modèles, pour les corriger, effec-

tuer des simulations, ou restituer l'apparence originale des objets, ce qui constitue aussi un intérêt majeur des modèles géométriques. Une autre question importante concerne alors le choix d'une représentation adaptée en fonction des traitements et des opérations d'édition à effectuer.

Dans cette thèse, notre objectif est d'apporter de la flexibilité à la chaîne de production de modèles 3D afin d'en réduire la complexité. Nous nous sommes concentrés sur deux étapes : la reconstruction de surface et l'édition du modèle. D'une part, nous avons étudié une approche dynamique de la reconstruction de surface dont le but est de construire directement des surfaces simplifiées à partir d'échantillons de points très denses et de pouvoir faire ensuite évoluer le résultat a posteriori en ajoutant ou en retirant des données. Ces travaux font l'objet de la première partie de la thèse. D'autre part, nous avons mené une réflexion sur un modèle géométrique hybride tirant parti de représentations de surface volumique et par frontière. Ce modèle est utile pour réaliser des opérations sur les modèles reconstruits. Ces travaux font l'objet de la seconde partie de la thèse, qui peut être considérée comme indépendante de la première.

Dans la suite de ce chapitre, nous commençons par rappeler le processus classique de création de modèles géométriques 3D à partir d'objets physiques (section 1). Ensuite, nous détaillons la problématique, nos contributions et le plan associés à chacune des deux parties de la thèse (sections 2 et 3).

1 Création de modèles 3D à partir d'objets physiques

La figure 1 décrit de manière synthétique le processus standard de création d'un modèle d'un objet physique à l'aide d'un dispositif de type scanner laser. Ce processus comporte trois étapes fondamentales. La première étape d'acquisition consiste à effectuer des mesures de la géométrie de la surface de l'objet sous différents angles pour en tirer des données en trois dimensions, le plus souvent sous la forme d'une collection d'images de profondeur correspondant chacune à un point de vue d'acquisition. Chaque image de profondeur consiste en une grille régulière de pixels auxquels sont associées des coordonnées 3D calculées en fonction de la distance qui sépare le capteur de la surface. L'étape de recalage consiste à aligner les images de profondeur dans un même repère pour ensuite les fusionner, de manière à obtenir un nuage de points répartis sur toute la surface numérisée de l'objet. La dernière étape fondamentale est celle qui consiste à reconstituer la surface de l'objet numérisé à partir de l'échantillon de points. Cette étape de reconstruction de surface est au centre de la problématique de la première partie de cette thèse. Le modèle obtenu peut ensuite subir divers traitements géométriques, comme la simplification, le lissage ou la compression. Une autre commodité consiste à pouvoir modifier et combiner de tels modèles pour les enrichir, les réparer, ou effectuer d'autres simulations, ce qui constitue le thème de la seconde partie de la thèse.

Dans les paragraphes suivants, nous présentons brièvement ces différentes étapes afin de donner au lecteur une vision plus précise de la complexité de cette chaîne de production et du besoin de flexibilité, en soulignant les aspects pertinents dans le cadre de cette thèse. Par la suite, nous décrirons plus en détail les méthodes de reconstruction de surface et les techniques d'édition de modèle évoquées ici.

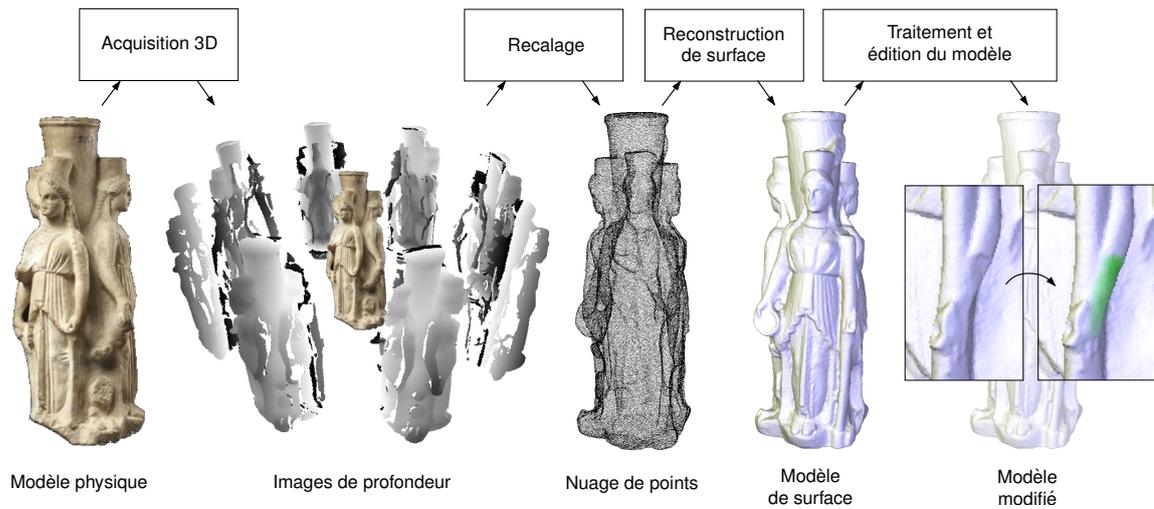


FIG. 1 – Processus classique de création d'un modèle 3D à partir d'un objet physique numérisé.

Acquisition 3D La création de modèle d'objets physiques nécessite de pouvoir mesurer la géométrie de la surface. La technologie des systèmes d'acquisition 3D a progressé de façon significative ces dernières années avec le développement de capteurs numériques de grande précision. Les scanners par triangulation sont les plus répandus dans le contexte de la numérisation d'objets d'art [20, 103]. Ceux-ci projettent une grille sur la surface de l'objet, soit sous la forme d'un faisceau laser balayant la surface [127], soit sous la forme d'une lumière structurée. Connaissant la position et la direction de la source de lumière et du capteur, les positions 3D des points éclairés peuvent être retrouvées facilement. Plusieurs points de vue sont généralement nécessaires pour couvrir la totalité de la surface d'un objet. Les données sont alors recueillies sous la forme d'images de profondeur. Selon les dispositifs, différents attributs peuvent être associés à chaque point, comme les normales, la couleur, la réflectance ou la précision des mesures.

À l'issue du processus d'acquisition, les images de profondeur correspondant aux différents points de vue nécessitent d'être alignées dans un même repère. Le principal algorithme de recalage d'images de profondeur est l'algorithme ICP (*Iterative Closest Point*) [2, 55, 142]. Étant donné deux images qui se recouvrent partiellement, cet algorithme a pour but de déterminer une transformation rigide alignant au mieux les deux images par rapport à une mesure d'erreur. L'alignement est corrigé de manière itérative selon un principe de descente de gradient jusqu'à atteindre un optimum local. Il existe ensuite diverses stratégies pour effectuer le recalage global [2, 100].

La numérisation 3D pose des problèmes de planification pour déterminer les points de vues qui permettent de couvrir toute la surface d'un objet [114]. Selon la complexité de la géométrie des surfaces à numériser, ainsi que l'encombrement et le nombre de degrés de liberté du dispositif d'acquisition, il peut être difficile de capturer certaines régions. Des auto-occlusions peuvent également conduire à des modèles incomplets. Les données acquises comportent aussi toujours une forme de bruit, liée au processus physique de mesure. La précision des mesures peut aussi varier en fonction de l'angle d'incidence de la source lumineuse et des propriétés optiques des matériaux [102]. Le bruit peut se manifester soit sous la forme de points aberrants, soit sous la forme de petites déviations des

points par rapport à la surface de l'objet (bruit *gaussien*). Des prétraitements sont ainsi souvent nécessaires pour construire un modèle de surface « valide » à partir d'un ensemble de mesures physiques. Par exemple, des techniques de lissage peuvent être appliquées pour réduire le bruit [168, 188]. Une technique de diffusion volumétrique a été développée pour reconstituer les données manquantes dans les régions où la surface n'a pas pu être numérisée [68].

Reconstruction de surface L'étape suivante de la chaîne de création d'un modèle 3D consiste à convertir le nuage de points en une surface, par exemple représentée sous la forme d'un maillage polygonal ou d'une surface implicite. Ce processus de reconstruction de surface fait l'objet de travaux de recherche en informatique graphique, en modélisation géométrique et en géométrie algorithmique depuis plus de vingt ans [140, 51, 169]. L'objectif de la reconstruction de surface est de produire un modèle de surface continu à partir du nuage de points, respectant autant que possible les propriétés géométriques et topologiques de la surface de l'objet numérisé. La difficulté de retrouver la géométrie et la topologie de la surface originale dépend des propriétés de cette surface, du modèle utilisé, et de l'échantillon. En particulier, la densité, la redondance, le bruit dans les données, la continuité et les bords d'une surface font de la reconstruction un problème complexe.

Les techniques de reconstruction de surface peuvent être classifiées en deux approches principales : l'approche combinatoire et l'approche par ajustement d'un modèle prédéfini. Les méthodes combinatoires ont pour principe d'établir des relations de connectivité entre les points d'un échantillon, en s'appuyant parfois sur les propriétés de structures géométriques permettant de modéliser des relations de proximité dans l'espace. Les méthodes de reconstruction de surface par ajustement de modèle cherchent quant à elles à contraindre un modèle mathématique de surface, global ou local, déterminé a priori, pour minimiser l'erreur entre le modèle et les données.

Traitement et édition du modèle Les modèles reconstruits ne sont pas toujours dans une forme définitive. Selon les applications visées, ceux-ci peuvent en effet être trop détaillés, ou bien leur représentation inadaptée. En outre, ces modèles peuvent servir de base à la création d'autres modèles ou constituer un support de simulation, par exemple pour de la restauration virtuelle. On peut ainsi distinguer deux types d'opérations : les traitements géométriques et les opérations d'édition, qui constituent deux domaines de recherche très vastes.

Les traitements peuvent consister à réduire la complexité d'un modèle maillé en appliquant des algorithmes de simplification de maillage, éventuellement multirésolution [99, 132, 131, 61], ou en appliquant des algorithmes de compression pour réduire le coût de stockage des modèles [10]. D'autres types de traitements relèvent de l'analyse de la surface, du filtrage, du rééchantillonnage, de la segmentation, ou encore de la conversion de représentation.

L'édition des modèles concerne la modification de leur forme ou de leur apparence, avec généralement un aspect interactif. Les techniques d'édition de modèles sont très diverses. On distingue principalement les techniques de combinaison de modèles, de déformation, et les techniques d'application de textures 2D ou 3D.

2 Reconstruction dynamique de surface

Dans la première partie de cette thèse, nous nous intéressons au problème de la reconstruction de surface à partir d'échantillons de points non structurés.

Les données issues des dispositifs d'acquisition sont généralement redondantes d'un point de vue géométrique. Cette redondance est intrinsèque au processus d'échantillonnage des surfaces sur des grilles régulières. En outre, le niveau de précision peut être mal approprié selon le contexte dans lequel ce modèle sera utilisé. La chaîne de traitement classique consiste à reconstruire une surface très détaillée, puis à simplifier cette dernière, ou à en construire une représentation à niveaux de détail au moyen d'algorithmes de simplification de surfaces maillées. Les techniques combinatoires de reconstruction de surface ont l'avantage de produire une surface triangulée qui peut être visualisée directement à l'issue du processus de reconstruction. Cette représentation sert aussi de base à de nombreux algorithmes de traitement géométriques (lissage, compression). Cependant, le fait de prendre en compte toutes les données dans la phase de reconstruction et dans la phase de simplification peut rendre celles-ci inutilement coûteuses si des relations de connectivité cohérentes doivent être établies pour tous les points.

Une alternative consiste à recourir à des techniques de simplification d'échantillons de points avant d'appliquer un algorithme de reconstruction de surface. Une réduction préalable de la complexité des données permet en effet de manipuler des structures combinatoires plus légères et d'accélérer les traitements. Ces techniques de simplification d'échantillons de points peuvent être assimilées à des techniques de simplification de surface particulières, à ceci près qu'on ne dispose pas de représentation explicite de la surface. Certaines propriétés locales de la surface doivent donc faire l'objet d'estimations. La simplification d'échantillons de points met ainsi en œuvre des structures et des mécanismes impliqués dans la reconstruction de surface, bien que l'objectif soit différent. Par conséquent, il est naturel d'envisager d'intégrer simplification et reconstruction de surface à partir d'échantillons de points au sein d'un même cadre de travail. Cette piste de recherche a déjà fait l'objet de plusieurs travaux depuis cinq ans, qui se sont intéressés à la génération de surfaces maillées simplifiées. Cependant, les résultats produits par ces méthodes restent figés, la surface reconstruite ne pouvant pas être mise à jour facilement. Or, il peut être intéressant de pouvoir faire évoluer ce résultat si les données ne sont pas toutes disponibles en même temps, ou pour faire face à des contraintes de transmission sur un réseau. Il est donc intéressant de disposer d'une méthode flexible de reconstruction de surface, capable d'analyser la pertinence des données et de mettre à jour le résultat dynamiquement.

Dans cette thèse, nous proposons un cadre de travail combinatoire dynamique pour la reconstruction de surface, dans lequel la reconstruction devient *sélective* et *évolutive*. Notre objectif est de reconstruire directement une surface triangulée simplifiée à partir d'un échantillon de points dense en faisant coopérer reconstruction et simplification, en offrant ensuite la possibilité d'ajouter ou de retirer des données dynamiquement (fig. 2). Notre système dynamique de reconstruction de surface est de nature combinatoire tout en faisant intervenir des estimations locales de propriétés de la surface. Il est constitué de deux modules principaux : un algorithme de reconstruction sélective et un algorithme de mise à jour dynamique.

- L’algorithme de reconstruction sélective permet de reconstruire directement une surface triangulée simplifiée à partir d’un échantillon de points dense en éliminant les données redondantes relativement à une mesure géométrique d’épaisseur locale.
- L’algorithme de mise à jour dynamique permet d’adapter a posteriori la reconstruction localement suite à une modification locale du niveau de précision, et plus généralement suite à l’insertion ou au retrait de données.

Notre système repose sur l’algorithme de convection géométrique introduit par Chainé [52], que nous avons étendu. Cet algorithme offre plusieurs propriétés intéressantes pour faire coopérer reconstruction et simplification des données d’une part, et pour mettre à jour le résultat dynamiquement d’autre part. Pour simplifier les données tout en restant dans des conditions d’échantillonnage favorables à l’algorithme de reconstruction, nous proposons une mesure d’erreur géométrique qui repose sur une estimation locale de l’épaisseur du solide borné par la surface échantillonnée. Deux applications importantes de la mise à jour dynamique que nous présentons sont la correction interactive du niveau de précision de la reconstruction et la reconstruction sélective à partir d’échantillons de points qui ne tiennent pas entièrement en mémoire centrale.

Dans le chapitre 1, nous présentons une étude bibliographique des travaux réalisés en reconstruction et de surface et en simplification d’échantillons de points. Le chapitre 2 présente l’algorithme de reconstruction de surface par convection géométrique sur lequel s’appuie notre système. Dans le chapitre 3, nous proposons une mesure d’épaisseur locale d’un solide sur laquelle nous appuyons notre technique de simplification. Nous décrivons notre algorithme de reconstruction sélective dans le chapitre 4, puis nous présentons l’algorithme de mise à jour dynamique dans le chapitre 5, avec son application à la correction interactive. Dans le chapitre 6, nous présentons enfin une extension de notre algorithme de reconstruction sélective selon un principe de subdivision-fusion pour traiter des données sans les charger entièrement en mémoire centrale, avant de conclure sur ces travaux dans le chapitre 7.

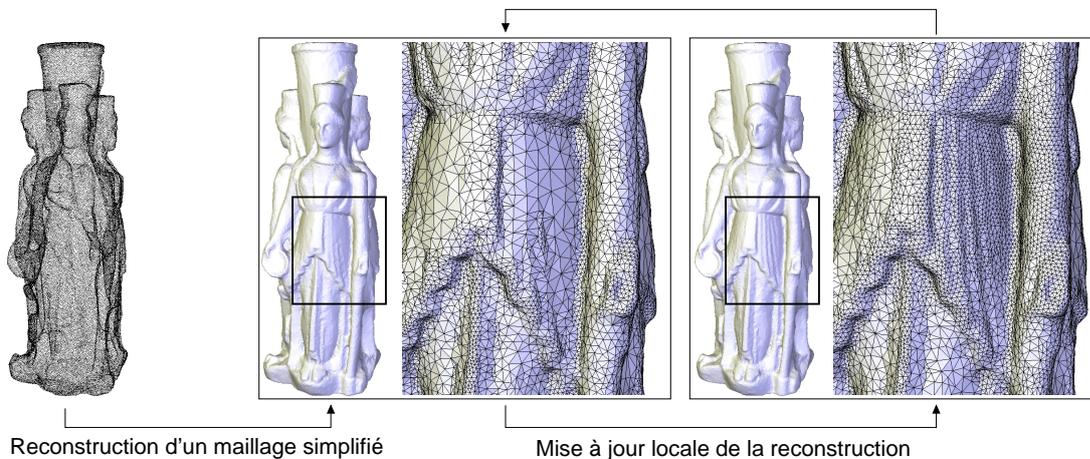


FIG. 2 – Notre système dynamique de reconstruction de surface.

3 Modélisation multireprésentation

La seconde partie de la thèse est consacrée à l'étude d'un modèle géométrique hybride destiné à la création de formes complexes à partir d'objets numérisés ou de primitives simples.

Une question fondamentale dans le cadre de l'édition interactive de modèles 3D concerne le choix de la représentation de surface sous-jacente. Par le passé, une grande diversité de représentations géométriques ont été développées, chacune présentant des propriétés et des limites propres. Selon les formes à modéliser et les opérations à effectuer, il existe ainsi des représentations plus adaptées que d'autres. Pour pouvoir bénéficier des avantages de différents modèles, il est généralement nécessaire de passer par des conversions globales coûteuses et souvent irréversibles. Une autre approche consiste à utiliser des méthodes ou manipuler des représentations hybrides. Notre objectif est de s'affranchir de ce type de restriction en faisant coopérer plusieurs représentations au sein d'un système de modélisation unifié permettant de tirer parti de leurs avantages complémentaires de manière transparente pour l'utilisateur.

Au niveau le plus abstrait, un objet solide peut être représenté de deux manières différentes : soit de manière volumique, soit par sa frontière, comme une surface 2D. Ces deux types de représentations se complètent par la nature de l'information qu'elles portent sur un modèle 3D. Nous avons ainsi choisi d'axer notre étude sur les *surfaces implicites* et les *maillages polygonaux*.

Les surfaces implicites [24, 186] constituent un modèle volumique puissant pour représenter des surfaces lisses, recommandé pour modéliser des formes organiques comme celles que l'on peut retrouver dans des sculptures. Une surface implicite est définie comme l'ensemble des points \mathbf{p} de l'espace tels que $f(\mathbf{p}) = 0$, où f désigne la fonction qui caractérise la surface $S = \{\mathbf{p} \in \mathbb{R}^3 | f(\mathbf{p}) = 0\}$. Cette représentation est particulièrement bien adaptée aux opérations booléennes de CSG (*Constructive Solid Geometry*) et de *mélange*, caractérisé par un raccordement lisse, et peuvent être déformées par transformation de l'espace. Les surfaces implicites changent également de topologie facilement, et la consistance de la surface est garantie par construction. Cependant, du fait que la surface n'est pas connue explicitement, le contrôle de ces surfaces est peu intuitif, elles se prêtent mal aux déformations locales, et leur visualisation nécessite un échantillonnage qui peut être coûteux pour des modèles complexes.

Les maillages polygonaux représentent les objets par une approximation linéaire par morceaux de leur frontière sous la forme d'un réseau cohérent de sommets, arêtes et faces. Cette représentation est devenue un standard pour l'échange et la visualisation interactive de modèles 3D complexes, optimisée par les cartes graphiques actuelles. Un grand nombre d'outils permettant de manipuler ces surfaces interactivement ont été développés, tels que les FFD (*Free-Form Deformations*) [174] qui fournissent un contrôle local intuitif de la géométrie. La possibilité de paramétrer une surface maillée est aussi particulièrement intéressante pour l'application de textures par exemple. Toutefois, la combinaison de maillages est une tâche complexe, ils sont peu adaptés au mélange, et il est difficile de garantir la validité topologique de la surface sous-jacente dans leur manipulation.

Dans cette thèse, nous proposons un modèle géométrique tirant parti de la complémentarité entre la représentation implicite et la représentation par maillage polygonal. Ce modèle est caractérisé par une

structure d'arbre dont les feuilles sont des primitives et les nœuds des opérations exploitant la représentation qui leur est la plus adaptée. Les primitives que nous manipulons peuvent être des surfaces implicites à squelette, des maillages triangulaires, ou bien des modèles représentés par des échantillons de points. Ces modèles sont combinés à l'aide d'opérations booléennes, de mélange et de déformation, incluant des FFD (fig. 3). Nous appelons notre modèle l'HybridTree, qui peut être vu comme une généralisation du modèle du BlobTree [191]. Nous évaluons cette structure à la volée à travers trois types de requêtes fondamentales : évaluation de la fonction potentiel et du gradient, classification spatiale (*Point Membership Classification*), et maillage. L'originalité de notre modèle réside dans le système d'évaluation, qui passe dynamiquement d'un modèle de surface à un autre de manière à utiliser le plus adapté pour chaque type d'opération, de manière transparente pour l'utilisateur. Le cœur de notre système est une représentation duale implicite/maillage pour chaque nœud mettant en œuvre des conversions locales. Les différents types de nœud de l'HybridTree sont évalués automatiquement de manière spécifique et optimisée en fonction de la formulation nécessaire à chaque type d'opération. Pour manipuler des modèles sous la forme d'échantillons de points, nous utilisons l'algorithme de reconstruction sélective développé dans la première partie de la thèse.

Le chapitre 1 présente plus en détails les deux représentations de surface impliqués dans notre système, avec un état de l'art des opérations de combinaison et de déformation associées. Le chapitre 2 présente une étude bibliographique des méthodes multireprésentations combinant ces modèles. Dans le chapitre 3, nous décrivons l'HybridTree de façon détaillée et présentons quelques résultats expérimentaux, notamment un essai de restauration virtuelle, avant de conclure dans le chapitre 4.

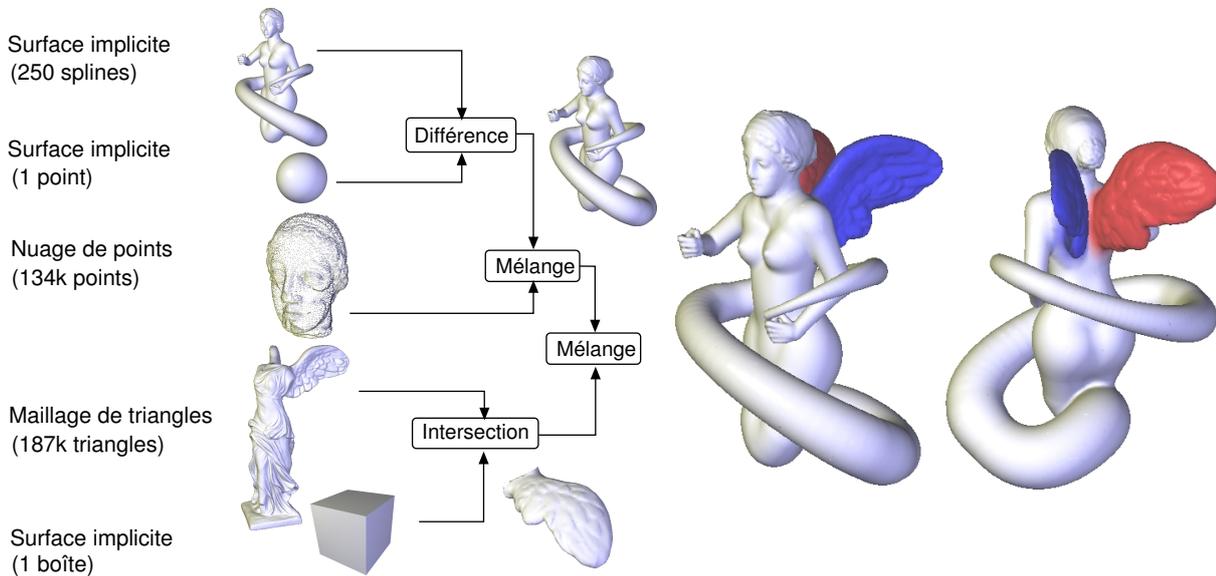


FIG. 3 – La structure d'HybridTree pour un modèle de Méduse ailée.

Première partie

Un cadre de travail dynamique pour la reconstruction de surface à partir d'échantillons de points non structurés

Dans le chapitre 1, nous rappelons le problème de la reconstruction de surface et présentons une étude bibliographique des principales techniques de reconstruction de surface et des techniques de simplification d'échantillons de points en mettant en évidence leur avantages et leur limites. Dans le chapitre 2, nous décrivons en détail l'algorithme de reconstruction de surface par convection géométrique, sur lequel reposent nos travaux. Le chapitre 3 introduit une mesure d'épaisseur locale d'un solide sur laquelle nous appuyons notre technique de simplification. Nous présentons ensuite notre algorithme de reconstruction sélective dans le chapitre 4, puis notre méthode de mise à jour dynamique et son application à la correction interactive dans le chapitre 5. Nous présentons enfin dans le chapitre 6 une extension de notre méthode de reconstruction sélective pour traiter des données qui ne tiennent pas entièrement en mémoire centrale. Nous concluons sur l'ensemble de ces travaux dans le chapitre 7.

État de l'art

1.1 Reconstruction de surface

Étant donné un ensemble de points $P \in \mathbb{R}^3$ échantillonnés sur une surface S , l'objectif de la reconstruction de surface est de calculer un modèle continu de S à partir de P . Ce modèle est appelé reconstruction de S à partir de P .

La reconstruction de surface intervient dans un grand nombre d'applications et fait l'objet de travaux de recherche depuis plus de vingt ans dans les communautés de l'informatique graphique, de la modélisation géométrique et de la géométrie algorithmique [140, 169]. Ce domaine a connu un essor important ces dernières années, avec le développement des technologies d'acquisition 3D. Dans ce chapitre, nous proposons une classification des différentes méthodes de reconstruction de surface existantes, en nous concentrant sur les techniques viables pour des données de grande taille. Nous tenterons de mettre en lumière leurs avantages et leurs limites vis à vis de la flexibilité que nous cherchons à atteindre.

Les techniques de reconstruction de surface peuvent être classifiées selon deux approches principales : l'approche combinatoire et l'approche par ajustement d'un modèle prédéfini. Un grand nombre de méthodes combinatoires ont pour principe d'établir des relations d'adjacence entre les points d'un échantillon [51, 72]. Elles ont pour la plupart été développées dans la communauté de la géométrie algorithmique, qui a apporté des contributions théoriques importantes au problème de la reconstruction de surface, notamment avec les notions d' ε -échantillon et de triangulation de Delaunay restreinte à une surface. Le lecteur pourra se reporter à l'annexe A de cette partie pour un rappel de ces notions, auxquelles il sera fait à nouveau référence par la suite.

La seconde approche repose sur l'idée d'approximer la surface échantillonnée à l'aide de modèles prédéfinis, sur la base d'hypothèses globales ou locales concernant la forme à reconstruire. Depuis 2001, on assiste à un fort engouement de la communauté graphique pour des approches implicites locales.

1.1.1 Approche combinatoire

Les méthodes combinatoires ont pour but d'établir des relations d'adjacence (ou de connectivité) entre les points voisins sur la surface échantillonnée. Pour atteindre cet objectif en l'absence de connaissances sur l'organisation des données, l'information topologique peut-être déduite de relations de proximité entre les points sous certaines hypothèses concernant la densité de l'échantillon. Les méthodes com-

binaires exploitent ainsi des structures géométriques permettant de mesurer la proximité des points dans l'espace. Parmi ces structures, le *diagramme de Voronoï* et son graphe dual, la *triangulation de Delaunay*, sont particulièrement populaires en géométrie algorithmique. Elles présentent en effet des propriétés intéressantes pour résoudre le problème de la reconstruction. En 1983, Edelsbrunner et al. [86] ont introduit le modèle des α -shapes en deux dimensions, établissant la première définition mathématique rigoureuse de la forme d'un échantillon de points dans le plan. En trois dimensions, Boissonnat [38] fut le premier à observer en 1984 que la triangulation de Delaunay d'un échantillon de points sur une surface peut contenir un sous-ensemble de facettes qui forme une bonne approximation de la surface échantillonnée. Beaucoup d'algorithmes en ont découlé, dont le principe général est de filtrer les relations de connectivité établies par la triangulation de Delaunay pour ne retenir que les pertinentes. Les conditions et les critères d'obtention d'une bonne approximation n'ont été précisés que beaucoup plus tard. À partir d'un théorème dû à Edelsbrunner et Shah [89], Amenta et Bern [13] ont prouvé en 1998 que si P est un ε -échantillon d'une surface lisse S avec $\varepsilon \leq 0.1$, alors la triangulation de Delaunay de P restreinte à S est homéomorphe à S . Bien sûr, dans le contexte de la reconstruction de surface, la surface S n'est pas connue. Ainsi, la majorité des algorithmes combinatoires reviennent à extraire une estimation plausible de la triangulation de Delaunay restreinte de la triangulation de Delaunay de l'échantillon de points. Cependant, les conditions d'échantillonnage ne sont pas toujours vérifiées en pratique, par exemple dans le cas de surfaces qui ne sont pas différentiables localement. Certains algorithmes identifient les facettes dont l'arête de Voronoï duale intersecte une approximation de la surface, en particulier à l'aide de plans tangents estimés, tandis que d'autres reposent sur l'idée que la surface à reconstruire sépare l'axe médian interne de l'axe médian externe d'un solide. Ces deux types d'information peuvent être obtenues à partir des pôles du diagramme de Voronoï.

Les relations de connectivité établies par la triangulation de Delaunay sont globales dans le sens où elles impliquent tous les points de l'échantillon. D'autres algorithmes combinatoires adoptent une approche plus locale en considérant la relation de plus proche voisin au sens de la distance euclidienne. Ces algorithmes ont recours à des structures de localisation spatiale adaptées pour obtenir de manière efficace les points les plus proches d'un point donné.

Dans les paragraphes suivants, nous allons présenter en détails les principales méthodes combinatoires, dont nous distinguons deux catégories principales en fonction de la nature de l'algorithme : les méthodes volumiques et les méthodes par expansion de surface. Les méthodes reposant sur la triangulation de Delaunay ayant recours aux pôles seront décrites dans un troisième temps.

1.1.1.1 Méthodes volumiques

Cette approche de la reconstruction de surface, la première reposant sur la triangulation de Delaunay, a été introduite par Boissonnat [38] avec l'algorithme *Sculpture*. À partir de l'enveloppe convexe de l'échantillon de points (les facettes externes de la triangulation de Delaunay), des cellules sont ouvertes itérativement en fonction d'un critère de priorité considérant la taille et la forme des facettes externes, ainsi que des règles topologiques. Durant le processus, la frontière externe des cellules restantes contient

de plus en plus de points, et l'algorithme s'arrête lorsque tous les points se trouvent sur celle-ci, ou lorsque certaines règles topologiques ne sont plus vérifiées. En particulier, ces règles limitent l'algorithme à la reconstruction de surfaces de genre nul. Un autre inconvénient majeur est que l'heuristique globale qui gouverne le processus rend le résultat dépendant de l'ordre dans lequel les cellules de la triangulation de Delaunay sont ouvertes. Dans certains cas, l'algorithme peut ainsi s'arrêter de manière prématurée, alors qu'une autre séquence d'opérations aurait pu conduire à un meilleur résultat [187]. Des algorithmes opérant selon le principe du processus de sculpture permettant des changements topologiques ont été proposés, mais toujours dépendants de l'ordre des opérations [23].

Plusieurs algorithmes s'appuyant sur la définition de relations d'ordre sur les cellules de la triangulation de Delaunay, plutôt que sur une heuristique globale fondée sur des critères locaux, ont été développés plus récemment. Le processus n'est alors plus soumis à des conditions topologiques, et le résultat ne dépend plus de l'ordre de traversée des cellules de la triangulation de Delaunay. L'algorithme *Wrap* d'Edelsbrunner [85], le *Flow Complex* introduit par Giesen et John [101] et l'algorithme de convection géométrique proposé par Chaine [52] entrent dans cette catégorie. Ces algorithmes s'intéressent au champ de gradient ∇d de la fonction de distance $d : \mathbb{R}^3 \rightarrow \mathbb{R}, \mathbf{x} \mapsto d(\mathbf{x}) = \min_{\mathbf{p} \in P} \|\mathbf{x} - \mathbf{p}\|$ induite par un échantillon de points P sur une surface, et au rapport que cette fonction entretient avec le positionnement relatif du diagramme de Voronoï de P et de sa triangulation de Delaunay duale. Dans l'algorithme *Wrap*, Edelsbrunner définit une *relation de flot* sur l'ensemble des cellules de la triangulation de Delaunay de l'échantillon à partir du champ de gradient ∇d . La surface reconstruite est un sous-ensemble de facettes de la triangulation de Delaunay formant le bord de l'union des *variétés stables* d'un sous-ensemble de maxima de d , constituées de l'ensemble des points qui convergent vers ces maxima suivant le flot induit par ∇d . Le *Flow Complex* est une structure récursive de variétés stables qui permet d'obtenir une reconstruction similaire à celle de l'algorithme *Wrap*. Dey et al. [80] ont récemment prouvé qu'il est possible de produire une reconstruction correcte à partir du *Flow Complex* sous l'hypothèse que P est un (ε, δ) -échantillon avec $\frac{\delta}{\varepsilon}$ fixé.

Le processus de sculpture peut aussi être interprété comme l'évolution d'une surface à travers la triangulation de Delaunay de l'échantillon de points. Dans l'algorithme de convection géométrique [52], celui-ci est guidé par un modèle physique de convection dont l'intérêt pour la reconstruction a été mis en évidence par Zhao et al. dans [198]. L'algorithme consiste à contracter une surface englobant l'échantillon de points sous l'influence du champ de gradient $-\nabla d$ jusqu'à atteindre un état stable s'appuyant sur la triangulation de Delaunay et caractérisé par une *propriété de Gabriel orientée*, qui signifie que pour toutes les facettes de la surface reconstruite, la demi-sphère diamétrale orientée vers l'intérieur de la surface est vide de points de l'échantillon. Une caractéristique importante de cette dernière approche est la possibilité de maintenir une structure de surface consistante (2-variété combinatoire) à chaque étape du processus de sculpture. À la base de notre technique de reconstruction sélective et dynamique, la méthode de convection géométrique sera exposée plus en détail dans le chapitre suivant.

1.1.1.2 Méthodes de reconstruction par expansion de surface

Les méthodes par expansion de surface reconstruisent une surface triangulée de manière incrémentale à partir des bords d'une surface initiale. La surface peut être construite en considérant directement un critère topologique global, ou bien à l'issue d'une phase de post-traitement d'un ensemble de facettes.

Dans [38], Boissonnat a proposé une seconde méthode reconstruisant une surface triangulée à partir d'un échantillon de points de manière incrémentale en faisant croître le maillage à partir de facettes-germes. Un contour est propagé à partir des bords de la surface courante en considérant des propriétés géométriques locales pour déterminer les nouveaux triangles à former. Étant donné une arête du contour, les points de son voisinage sont projetés sur un plan tangent estimé localement à l'une de ses extrémités. La facette suivante est construite à partir de cette arête et du point de ce voisinage qui maximise l'angle au sommet de la facette opposé à l'arête lorsque celle-ci est projetée sur le plan tangent. L'opération est répétée jusqu'à ce que la surface soit fermée, et tous les points pris en compte. De nouvelles facettes-germes peuvent être créées si la surface à reconstruire comporte plusieurs composantes connexes. Cette méthode n'utilise pas la triangulation de Delaunay, mais l'auteur remarque que les facettes pourraient être choisies dans la triangulation de Delaunay. Cet algorithme reconstruit une surface consistante sous certaines conditions topologiques restrictives.

Plusieurs autres algorithmes adoptant ce principe de construction incrémentale par avancée de front ont été développés. L'algorithme *Ball-pivoting* [33], inspiré par le modèle des α -shapes [88], construit la surface de manière incrémentale en faisant « rouler » une boule sur les points. La surface est initialisée à partir d'une facette-germe orientée telle qu'une boule de rayon fixé passant par ses trois sommets ne contient aucun autre point. La boule pivote autour d'une des arêtes de la facette dans le sens de sa normale jusqu'à ce qu'elle touche un autre point, formant ainsi une nouvelle facette. L'opération est répétée pour toutes les arêtes du bord de la surface courante. Une fois qu'elles ont toutes été visitées, de nouveaux triangles-germes peuvent être formés si tous les points n'ont pas été atteints. L'algorithme produit une triangulation correcte si l'échantillon de points est uniforme. Pour traiter des échantillons de densité variables, plusieurs passes sont nécessaires, en faisant varier le rayon de la boule. L'algorithme nécessite aussi que les points soient équipés de normales orientées de manière cohérente. Un algorithme plus robuste filtrant la triangulation de Delaunay de l'échantillon de points et ne nécessitant pas les normales a été ensuite proposé par Cohen-Steiner et Da [62]. Le critère de croissance repose sur des mesures de distances et d'angles pour choisir la facette la plus plausible à chaque étape. Ces algorithmes assurent par construction que la surface reconstruite est une 2-variété combinatoire, mais n'offrent pas de garantie sur la qualité de l'approximation.

Un inconvénient du maintien d'un critère topologique global tout au long du processus d'expansion est le risque de faire échouer la reconstruction en cas d'erreur. Une autre approche consiste à appliquer le processus à un sous-ensemble de facettes préalablement construites ou sélectionnées d'après certains critères géométriques, sans que celles-ci ne forment nécessairement une surface cohérente. Le processus d'expansion de surface consiste alors à extraire la surface reconstruite de ce sous-ensemble.

Ohtake et al. [156] proposent d'effectuer le processus d'expansion de surface en formant les facettes duales d'une couverture de l'échantillon de points avec des sphères. La première étape de la méthode consiste à construire une couverture de sphères dont le rayon s'adapte à la courbure locale, avec pour chacune d'elle le calcul d'un point représentatif du voisinage de points qu'elle contient. Des relations d'adjacence sont ensuite établies entre les points représentatifs en examinant les intersections entre les sphères de la couverture, puis un maillage de la surface est obtenu en filtrant ces relations de manière à ce que les facettes incidentes à un sommet forment un disque topologique plausible. La technique de rééchantillonnage mise en œuvre dans cette méthode sera détaillée dans la section 1.2.1.

Gopi et al. [104] approximent la surface dans un voisinage autour de chaque point de l'échantillon par une triangulation 2D locale. L'algorithme procède ainsi en trois étapes pour chaque point $\mathbf{p} \in P$. Tout d'abord, un plan tangent orienté est estimé localement en \mathbf{p} , puis les normales ont été utilisées pour estimer localement les courbures principales. Un voisinage de points autour de \mathbf{p} est ensuite sélectionné, comportant tous les points situés dans une boule dont le rayon est calculé en fonction des courbures estimées. Les points de ce voisinage sont projetés sur le plan tangent de \mathbf{p} , puis une triangulation de Delaunay 2D des points du voisinage projeté permet de construire des triangles candidats incidents à \mathbf{p} . Trois points de l'échantillon forment une facette de la surface reconstruite s'ils sont tous mutuellement contenus dans leurs voisinages de Delaunay 2D. La surface est ensuite extraite par expansion de surface. L'algorithme produit une surface triangulée homéomorphe à S sous l'hypothèse que l'échantillon de points est suffisamment dense relativement à la courbure locale, et qu'une condition d'uniformité locale soit vérifiée. Pour tenir compte des différentes parties de la surface S qui sont proches dans le calcul des rayons de leurs voisinages, les auteurs proposent de mesurer l'épaisseur d'un voisinage avec un critère similaire à celui du rayon topologique intervenant dans notre mesure d'épaisseur locale.

Adamy et al. [4] ont proposé un algorithme de filtrage de la triangulation de Delaunay reposant sur une extension du complexe de Gabriel. Le complexe de Gabriel est un sous-ensemble de la triangulation de Delaunay qui contient les facettes dont la sphère circonscrite diamétrale est vide de points de l'échantillon. Les auteurs définissent un ordre sur les facettes de Gabriel relativement à la plus grande boule vide passant par leurs sommets. Une facette a une plus grande priorité et est davantage susceptible d'appartenir à la surface reconstruite si elle possède un côté où les points sont éloignés. L'algorithme de sélection de facettes consiste à construire un disque topologique en chaque point de l'échantillon à partir de l'ensemble ordonné de facettes de Gabriel. Cet ensemble de facettes est ensuite filtré pour en extraire une surface. Aucune garantie n'est fournie sur le résultat.

1.1.1.3 Méthodes ayant recours au pôles

Les algorithmes décrits jusque là reposent essentiellement sur des critères locaux pour établir ou sélectionner des relations de connectivité valides. Une famille d'algorithmes exploite le fait qu'un objet solide peut être vu comme une union infinie de boules maximales centrées sur son axe médian. Pour un échantillon de points suffisamment dense sur une surface bornant un solide, les pôles du diagramme de Voronoï sont proches de l'axe médian de ce solide. Cette information de nature globale a été exploitée

à la fois dans le cadre d'algorithmes volumiques et dans le cadre d'algorithmes mettant en œuvre un mécanisme d'expansion de surface.

Power Crust L'algorithme Power Crust proposé par Amenta et al. [15] repose sur l'observation que tout point d'une surface compacte échantillonnée est sur la frontière de deux boules médiales : la boule externe à la surface et la boule interne. Les boules médiales peuvent être approximées par les boules circonscrites aux cellules de Delaunay duales d'un pôle, appelées *boules polaires*. Amenta et al. ont développé des critères locaux pour distinguer les boules polaires de natures différentes et une heuristique globale pour distinguer les internes et les externes. Si deux boules polaires adjacentes s'intersectent profondément, alors les deux pôles sont considérés comme étant du même côté de la surface. Inversement, si les deux boules s'intersectent peu profondément, alors l'une des deux cellules est interne, et l'autre externe. L'algorithme Power Crust construit le *diagramme de puissance* de ces boules. La surface reconstruite est l'ensemble des facettes de ce diagramme dont les arêtes duales ont un sommet qui est un pôle interne et l'autre qui est un pôle externe.

Le principe de classification par analyse des boules polaires a été repris par la suite dans d'autres algorithmes. Dey et Goswami [82] ont mis en évidence que le critère d'angle d'intersection entre les boules polaires proposé dans Power Crust ne permet pas d'effectuer correctement la classification dans les régions bruitées. Ils montrent cependant que les boules polaires dans ces régions ont un rayon plus petit, et peuvent être détectées. Kolluri et al. [124] ont proposé un autre algorithme dérivé de Power Crust, appelé Eigen Crust, qui permet de reconstruire une surface fermée à partir d'un échantillon de points bruité comportant des points aberrants par partitionnement spectral du graphe des pôles. Mederos et al. [138] ont étendu l'algorithme Power Crust à des échantillons de points bruités selon un modèle de type gaussien, avec la garantie de produire une surface homéomorphe à la surface échantillonnée.

Interpolation des voisins naturels de fonctions de distance signées Boissonnat et Cazals [39] ont développé une méthode de reconstruction qui classe les pôles comme internes ou externes à la surface d'après le signe d'une fonction $f : \mathbb{R}^3 \rightarrow \mathbb{R}$ dont l'ensemble des *zéros* définit une surface implicite approximant la surface échantillonnée. La fonction f est construite par interpolation des voisins naturels de fonctions de distance *signées* définies autour de chaque point de l'échantillon comme la distance à son plan tangent, ce qui nécessite donc de disposer de vecteur normaux orientés de façon cohérente en chaque point. Les voisins naturels d'un point \mathbf{x} sont définis comme les voisins de \mathbf{x} dans la triangulation de Delaunay de $P \cup \{\mathbf{x}\}$. La coordonnée naturelle $\lambda_i(\mathbf{x}) \in [0, 1]$ associée à un point $\mathbf{p}_i \in P$ est définie par :

$$\lambda_i(\mathbf{x}) = \frac{w_i(\mathbf{x})}{\sum_i w_i(\mathbf{x})}$$

où $w_i(\mathbf{x})$ représente le volume de la portion de la cellule de Voronoï de \mathbf{p}_i « volée » par le point \mathbf{x} lorsqu'il est inséré dans le diagramme de Voronoï de P . La fonction f en un point \mathbf{x} est définie comme la distance

signée moyenne de ce point aux plans tangents des voisins naturels de \mathbf{x} :

$$f(\mathbf{x}) = \sum_{\mathbf{p}_i \in N(\mathbf{x})} \lambda_i(\mathbf{x}) d_i(\mathbf{x})$$

où $N(\mathbf{x})$ désigne l'ensemble des voisins naturels de \mathbf{x} et $d_i(\mathbf{x})$ la distance signée du point \mathbf{x} au plan tangent au point \mathbf{p}_i . Les plans tangents peuvent être estimés à partir des pôles, avec en plus une heuristique pour déterminer une orientation consistante [40]. De par les propriétés de continuité des coordonnées naturelles, l'ensemble des zéros de la fonction f définit une surface lisse \hat{S} interpolant les points de P , avec $f(\mathbf{x}) > 0$ à l'intérieur de la surface et $f(\mathbf{x}) < 0$ à l'extérieur. La surface reconstruite est alors constituée de l'ensemble des facettes de la triangulation de Delaunay de l'échantillon de points P restreinte à la surface \hat{S} . Une facette de la triangulation de Delaunay fait partie du résultat si et seulement si son arête de Voronoï duale $\mathbf{c}_1\mathbf{c}_2$ vérifie $f(\mathbf{c}_1)f(\mathbf{c}_2) < 0$.

Boissonnat et Cazals ont montré que l'ensemble des zéros de la fonction f converge vers la surface échantillonnée pour un ε -échantillon avec $\varepsilon \rightarrow 0$ et que la triangulation de Delaunay restreinte à \hat{S} est homéomorphe à la surface échantillonnée. La qualité de l'approximation linéaire obtenue peut être évaluée en examinant la distance de ses facettes à la surface implicite au sens de la fonction f . Elle peut être raffinée en ajoutant des points sur la surface implicite partout où la reconstruction est de qualité insuffisante. Boissonnat et Cazals [40] ont étendu leur méthode pour enrichir une surface reconstruite à partir d'un sous-échantillon tout en assurant que la reconstruction reste correcte (cette méthode sera décrite plus en détails dans la section 1.2.3).

Crust L'algorithme Crust [13, 12] considère l'enrichissement de la triangulation de Delaunay $D(P)$ de l'échantillon P avec l'ensemble des pôles Q du diagramme de Voronoï dual. Toutes les facettes de $D(P)$ traversées par l'axe médian sont détruites si les pôles sont insérés dans $D(P)$. Les facettes de la triangulation de Delaunay $D(P \cup Q)$ connectant trois points de P sont alors susceptibles d'appartenir à la triangulation de Delaunay restreinte à la surface. Cette collection de facettes ne forme en général pas la surface finale mais contient une surface qui est homéomorphe à S si l'échantillon P est un ε -échantillon avec $\varepsilon \leq 0.1$. La surface résultante est obtenue en appliquant un second processus de filtrage sur ces facettes candidates, commençant par éliminer les triangles dont la normale est trop éloignée des vecteurs pôles en leurs sommets. Les facettes restantes sont ensuite orientées de manière consistante, et une surface en est finalement extraite par expansion de surface. Il n'a pas été prouvé que le résultat de la phase d'extraction de surface est homéomorphe à la surface échantillonnée.

Cocone L'algorithme Cocone a été développé par Amenta et al. [14] à partir de l'algorithme Crust, en visant à simplifier ce dernier. De la même manière, l'algorithme consiste à calculer une collection de facettes candidates et à en extraire ensuite une surface. Les facettes candidates sont cette fois sélectionnées directement dans la triangulation de Delaunay de l'échantillon à l'aide d'un critère relatif à la géométrie des cellules de Voronoï. Le *cocône* d'un point \mathbf{p} de l'échantillon est défini comme l'ensemble des points $\mathbf{x} \in \mathbb{R}^3$ proches du plan tangent en \mathbf{p} au sens où le vecteur \mathbf{xp} forme un angle proche de $\frac{\pi}{2}$ avec la normale

en \mathbf{p} , qui peut être estimée par le vecteur pôle. Une facette est candidate si son arête de Voronoï duale intersecte les cocônes de ses trois sommets. Sous certaines conditions d'échantillonnage, une surface homéomorphe à la surface échantillonnée peut ensuite être extraite à partir des facettes candidates par expansion de surface. Le résultat de l'algorithme est une surface homéomorphe à la surface originale si l'échantillon est un ε -échantillon avec $\varepsilon \leq 0.08$.

La version originale de l'algorithme Cocone est très sensible aux conditions d'échantillonnage, qui sont rarement vérifiées en pratique. L'algorithme a été ensuite revisité à plusieurs reprises. Il a été utilisé pour détecter les régions sous-échantillonnées et les bords [74], puis étendu pour garantir la reconstruction de surfaces fermées [81]. Le même algorithme, associé à une méthode de détection de régions sur-échantillonnées [75] a été utilisé pour élaborer un algorithme de simplification d'échantillons de points par sous-échantillonnage [77] et un algorithme de simplification avec rééchantillonnage [79]. Nous reviendrons en détails sur ces algorithmes dans la section 1.2.2. Une version de l'algorithme Cocone capable de traiter des données de taille importante ne calculant pas la triangulation de Delaunay de l'ensemble des points en entrée mais de blocs de points a également été développée [78]. Une version de l'algorithme Cocone robuste au bruit a plus récemment été étudiée [82], avec des preuves reposant sur un modèle de bruit de type gaussien.

Dey, Funke et Ramos [73] ont élaboré un algorithme qui calcule le résultat de l'algorithme Cocone sans construire entièrement la triangulation de Delaunay des points de l'échantillon. Les auteurs supposent que l'échantillon est localement uniforme, ce qui implique que la plus petite sphère centrée sur un point de l'échantillon et incluant tous ses voisins dans la triangulation de Delaunay restreinte contient un nombre constant de points de P . Cette hypothèse permet d'obtenir les facettes candidates à partir d'une structure de recherche spatiale des plus proches voisins. En l'absence d'une triangulation de Delaunay, la normale à la surface en un point de l'échantillon est estimée à partir du triangle formé par ce point et ses deux plus proches voisins formant un angle suffisamment grand. Funke et Ramos [94] ont décrit une méthode permettant de simplifier un échantillon de points pour atteindre cette condition d'uniformité locale, mais l'algorithme n'a pas été implanté.

1.1.2 Ajustement de modèles

Les méthodes de reconstruction de surface par ajustement de modèles cherchent à contraindre un modèle mathématique de surface global ou local déterminé *a priori* pour minimiser l'écart entre le modèle et les données. Le processus de reconstruction se ramène alors principalement à un problème d'optimisation. La surface peut être contrainte à passer par les points de données, on parle d'*interpolation*, ou bien à proximité, on parle alors d'*approximation*.

Le modèle peut être défini comme une fonction dépendant d'un nombre fini de paramètres. L'objectif est alors de trouver les valeurs des paramètres pour que le modèle interpole ou approxime les données en entrée, selon une approche globale ou locale ; on parlera dans ce cas d'*ajustement de paramètres*. Il en était déjà question dans la section précédente lorsqu'il s'agissait d'estimer localement des propriétés différentielles de la surface échantillonnées, comme par exemple les normales, par des plans tangents, ou

les courbures, par des quadriques. Plus généralement, un modèle peut être représenté par une fonction de manière paramétrique ou implicite. Dans cette section, nous nous intéresserons spécifiquement aux modèles de surface implicite, qui présentent moins de contraintes et sont plus flexibles pour reconstruire des formes complexes selon une approche locale, en profitant de la capacité à les combiner facilement. L'aptitude d'un modèle à fournir une bonne approximation de la surface originale réside dans le choix de fonctions appropriées, qui suppose la formulation d'hypothèses sur les propriétés de la surface à reconstruire, en particulier de continuité. La nature de ces fonctions et celle de l'erreur à minimiser, ainsi que les conditions d'échantillonnage rendent le processus d'ajustement plus ou moins complexe et fiable. D'autres méthodes considèrent l'ajustement de *modèles déformable*, pour lesquels le problème de la reconstruction est résolu de manière itérative par déformations successives d'une surface initiale, continue ou discrète.

1.1.2.1 Reconstruction par ajustement d'un modèle de surface implicite

Le problème revient à interpoler ou approximer les données à l'aide d'une fonction $f : \mathbb{R}^3 \rightarrow \mathbb{R}$ continue dont l'ensemble des zéros représente une surface proche de la surface échantillonnée. Pour représenter des formes complexes, la fonction f est généralement construite comme une combinaison de fonctions simples. La surface à reconstruire peut être approximée globalement en combinant des fonctions permettant d'approximer la surface localement à un degré plus ou moins élevé, ou bien par une somme pondérée de fonctions de base radiales. Les méthodes locales ont généralement recours à des structures combinatoires pour décomposer les données dans l'espace ou établir des relations de proximité entre les points. De nombreuses méthodes reposent en effet sur l'estimation de propriétés différentielles de la surface, qui nécessitent de se doter d'une structure de voisinage sur l'échantillon de points. Le plus souvent, les relations de voisinage sont choisies comme des relations de plus proches voisins au sens de la distance euclidienne. Certaines méthodes par approximation locale construisent une fonction de distance à une surface implicite à partir de fonctions de distance à des plans tangents estimés localement en chaque point, tandis que d'autres construisent une surface polynomiale par morceaux en considérant localement la surface comme le graphe d'une fonction définie sur un domaine paramétrique plan.

Fonction de distance signée à des plans tangents Hoppe et al. [112] sont à l'origine de la première méthode de reconstruction de surface implicite proposée dans la communauté graphique. Ceux-ci estiment une fonction de distance signée $d : \mathbb{R}^3 \rightarrow \mathbb{R}$ entre un point $\mathbf{x} \in \mathbb{R}^3$ et la surface originale. L'ensemble des zéros de cette fonction $d(\mathbf{x}) = 0$ définit la surface reconstruite de manière implicite. La distance d'un point \mathbf{x} à la surface est définie comme la distance entre \mathbf{x} et le plan tangent associé au point de l'échantillon le plus proche. Le signe de la fonction de distance dépend alors du côté du plan tangent où \mathbf{x} se trouve. Le plan tangent en un point de l'échantillon est estimé aux moindres carrés par analyse en composantes principales des coordonnées de ce point et de ses voisins. La direction du vecteur normal est donnée par la direction de variance minimale de la position des points au sein de ce voisinage. L'orientation des plans tangents est déterminée dans un second temps à l'échelle globale de

manière à ce que les plans tangents associés à deux points proches aient une orientation similaire. La stratégie adoptée consiste à propager l'orientation des normales le long d'un arbre recouvrant minimum au sens de la distance euclidienne.

Les surfaces reconstruites sont de continuité C^0 . Cette technique d'estimation et d'orientation consistante des plans tangents a été reprise plus tard par diverses méthodes de reconstruction nécessitant de disposer d'une normale orientée en chaque point de l'échantillon.

Partition de l'unité L'approche de partition de l'unité consiste à décomposer le domaine des données en sous-domaines plus petits qui se recouvrent, où le problème de reconstruction est résolu localement. Dans un premier temps, les données sont approximées dans chaque sous-domaine séparément par des fonctions $f_i : \mathbb{R}^3 \rightarrow \mathbb{R}$, puis les solutions locales sont « mélangées » par une somme pondérée des approximations locales :

$$f(\mathbf{x}) = \sum_{i=1}^n w_i(\mathbf{x}) f_i(\mathbf{x})$$

où n représente le nombre de sous-domaines. Les fonctions $w_i : \mathbb{R}^3 \rightarrow \mathbb{R}$ sont des fonctions de poids à support compact, centrées sur les sous-domaines, dont la somme vaut un sur l'ensemble du domaine.

Une des premières méthodes de reconstruction de surface appliquant cette formulation est la méthode d'interpolation des voisins naturels de fonctions de distance signées de Boissonnat et Cazals [39] qui a été décrite dans la section 1.1.1.3. Les plans tangents définis en chaque point de l'échantillon correspondent aux reconstructions locales, et la triangulation de Delaunay de l'ensemble de points constitue la partition du domaine des données. Des méthodes mettant en œuvre des approximations de degré plus élevé ont été développées par la suite. Ohtake et al. [152] ont proposé de reconstruire la surface localement à l'aide de fonctions quadratiques ou quadratiques par morceaux capturant la forme locale de la surface. La partition est construite à partir d'une grille adaptative, une étude locale de la déviation des normales orientant le choix du modèle de surface dans chaque cellule-feuille. Cette méthode est très efficace sur des échantillons de points de taille importante et permet de contrôler le niveau de détail de la reconstruction. En revanche, elle est sensible à la qualité des données d'entrée, en particulier à celle des normales. Tobor et al. [181] ont utilisé l'approche de partition de l'unité avec des fonctions de base radiales pour calculer les approximations locales, avec une décomposition hiérarchique sur une grille. Cette approche permet de réduire la complexité du problème de reconstruction globale avec des RBF pour des données d'entrée de taille importante, technique qui sera décrite en détails par la suite.

Formulation par un opérateur de projection : *Moving Least Squares* Le modèle de surface MLS constitue une autre approche locale de reconstruction de surface implicite. La particularité de ce modèle est que la fonction représentant la surface globale n'est pas définie explicitement. Des approximations locales sont calculées uniquement au voisinage de points où on souhaite évaluer la fonction, et au moment de cette évaluation. Au cours de ces dernières années, cette technique a rencontré un certain succès dans la communauté graphique pour modéliser des surfaces à partir d'échantillons de points sans maintenir

une connectivité explicite.

Étant donné un échantillon de points orientés sur une surface, Levin [125, 126] a proposé un modèle de surface implicite approximant globalement les données fondé sur la méthode d'approximation locale de fonction des moindres carrés pondérés glissants, ou *Moving Least Squares* (MLS). Levin définit une *surface MLS* par l'intermédiaire d'un opérateur de projection $\Psi : B \rightarrow \mathbb{R}^3$ qui projette tout point situé dans un voisinage tubulaire B de cette surface sur la surface MLS elle-même. La surface MLS est ainsi définie comme l'ensemble des points de B qui se projettent sur eux-mêmes :

$$S_{MLS} = \{\mathbf{x} \in B \mid \Psi(\mathbf{x}) = \mathbf{x}\}$$

Dans cette approche, la fonction qui modélise la surface n'est pas déterminée explicitement ; elle est définie implicitement par l'opérateur de projection. Le cœur de la méthode est une procédure de projection en trois étapes qui calcule la projection d'un point $\mathbf{x} \mapsto \Psi(\mathbf{x})$ sur une approximation polynomiale locale de la surface dans un voisinage qui dépend de la position de $\Psi(\mathbf{x})$. La première étape consiste à déterminer un *plan de référence* qui sert ensuite de domaine paramétrique pour construire l'approximation polynomiale locale de la surface. Cette étape, la plus complexe, implique la résolution itérative d'un problème de minimisation non linéaire pour déterminer la direction de la normale en $\Psi(\mathbf{x})$. La deuxième étape consiste à calculer l'approximation polynomiale locale sur laquelle est projeté le point \mathbf{x} , ce qui se ramène à la résolution d'un problème standard de moindres carrés linéaires où les coefficients du polynôme sont déterminés en résolvant un système linéaire d'équations. La dernière étape consiste à calculer la position du point projeté sur la surface locale.

Levin a démontré que la surface résultante est une variété C^∞ sous certaines conditions de densité d'échantillonnage et de continuité déterminées par la fonction de poids impliquée dans les approximations locales. Cette technique est naturellement robuste au bruit gaussien, mais la procédure de projection peut échouer si la densité de l'échantillon de point est insuffisante par rapport à la géométrie locale, ou si celui-ci comporte des points aberrants.

Plusieurs auteurs ont revisité le modèle original de Levin pour fournir des algorithmes de reconstruction de surface robustes au bruit [195], multirésolution [139], ou capables de détecter des points aberrants et de représenter une surface lisse par morceaux [91]. Scheidegger et al. [170] ont proposé un algorithme spécifique pour trianguler une surface MLS selon la technique des *Marching Triangles*. Boissonnat et Oudot [43] ont également appliqué leur méthode d'échantillonnage de surface aux surfaces MLS pour produire une surface triangulée optimale en utilisant le modèle proposé par Adamson et al. [3].

Plusieurs auteurs se sont intéressés à la validité globale du modèle de surface MLS. Amenta et Kil [17] ont analysé la stabilité de l'opérateur de projection original en fonction de la proximité des points par rapport à la surface MLS. Ils ont également proposé une définition explicite des surfaces MLS en termes de points critiques d'une fonction d'énergie définie dans un champ vectoriel. Kolluri a prouvé la validité du modèle proposé par Adamson et al. [3] sous des conditions d'échantillonnage uniformes [123]. Une autre extension du modèle MLS proposée par Dey et al. [83] permet d'obtenir une

reconstruction correcte sous des conditions d'échantillonnage non-uniformes.

Fonctions de base radiales Les fonctions radiales de base, ou RBF pour *Radial Basis Functions* constituent une technique populaire d'interpolation de données par une fonction. Étant donné un ensemble de n points $\{\mathbf{p}_i\} \subset \mathbb{R}^3$ auxquels sont associées des valeurs scalaires $\{h_i\}$, l'objectif est d'approximer une fonction $f : \mathbb{R}^3 \rightarrow \mathbb{R}$ telle que $f(\mathbf{p}_i) \approx h_i$ pour tout (\mathbf{p}_i, h_i) en minimisant une erreur E choisie a priori, qui peut par exemple être de la forme :

$$E(f) = \sum_{i=1}^n (f(\mathbf{p}_i) - h_i)^2 + \lambda R(f)$$

où le premier terme correspond à une erreur aux moindres carrés, et le second, de régularisation, contrôle le comportement interpolant ($\lambda = 0$) ou approximant ($\lambda > 0$) de la solution en fonction d'une énergie de courbure R destinée à favoriser des solutions lisses, oscillant le moins possible.

Dans le cadre de l'approche RBF, la fonction qui minimise la fonctionnelle E s'exprime à partir d'une fonction de base $\phi : \mathbb{R}_+ \rightarrow \mathbb{R}$ à support local ou global dépendant du choix E , comme une combinaison linéaire

$$f(\mathbf{x}) = \sum_{j=1}^m w_j \phi(\|\mathbf{x} - \mathbf{c}_j\|)$$

où les points $\{\mathbf{c}_j\}$ représentent un ensemble de m points de contrainte, ou centres, et les coefficients $\{w_j\}$ des poids attribués à chaque fonction de base. Le problème de reconstruction de surface revient alors à déterminer le vecteur $\mathbf{w} = \{w_1, \dots, w_m\}$, qui s'obtient en résolvant un système linéaire d'équations.

L'ensemble des zéros de la fonction f définit une surface implicite qui interpole ou approxime les centres. Pour un échantillon de points quelconque sur une surface, tous les points sont associés à des contraintes $f(\mathbf{p}_i) = 0$. Pour éviter la solution triviale $\mathbf{w} = 0$, des points de contrainte supplémentaires à valeurs non nulles doivent être choisis à l'extérieur et/ou à l'intérieur de la surface. Ces contraintes peuvent par exemple être obtenues par décalage le long de normales orientées.

Les premiers travaux dans la communauté graphique sur cette approche de la reconstruction de surface ont été réalisés par Savchenko et al. [167], puis Turk et O'Brien [182, 183]. Les méthodes utilisant des RBF à support global sont particulièrement robustes, mais le système linéaire d'équations à résoudre est dense, ce qui rend difficile le traitement d'échantillons de plus de quelques dizaines milliers points (complexité en $O(n^3)$). Pour accélérer l'évaluation et traiter des données de taille plus importante, différentes approches ont été étudiées. Carr et al. [50] réduisent le nombre de centres par une procédure de sélection itérative et utilisent une technique d'évaluation rapide des RBF (*Fast Multipole Method*) dont le principe est d'évaluer les fonctions de base de manière approximative au delà d'une certaine distance aux centres. Morse et al. [147] et Ohtake et al. [154] ont proposé d'utiliser des RBF à support compact, pour lesquelles le système d'équations devient creux et peut être résolu efficacement. Cependant, le choix de rayons de support adaptés est difficile dans le cas d'échantillons non uniformes. Ce type de solution pose aussi un problème de scalabilité pour traiter de gros volumes de données. Samozino et al. [165] ont ré-

cement proposé une nouvelle méthode de reconstruction de surface à partir de RBF à support compact dépassant ces limites en utilisant un sous-ensemble des pôles du diagramme de Voronoï de l'échantillon de points en entrée comme centres. Les auteurs s'appuient sur le principe d'approximation d'un objet solide par une union de boules maximales centrées sur son axe médian interne, exploité dans plusieurs méthodes combinatoires. Chaque centre est associé à une boule maximale vide, dont le rayon est utilisé pour adapter le support de chaque fonction de base. En outre, le nombre de centres peut être contrôlé en filtrant le sous-ensemble des pôles à l'aide de la notion de λ -axe médian [53]. Ce travail représente une association intéressante entre une approche combinatoire et une approche modèle.

Une autre manière de s'attaquer aux problèmes de performance de l'approche RBF consiste à procéder par partition de l'unité. Tobor et al. [181] et Ohtake et al. [155] décomposent ainsi les données de manière hiérarchique en sous-domaines où le problème d'interpolation est résolu localement. Les reconstructions locales sont mélangées au niveau de chaque nœud de la hiérarchie, la racine contenant la fonction correspondant à la reconstruction globale.

Malgré les techniques de réduction de centres et l'usage de méthodes numériques accélérées, les performances de cette approche, globale par nature, restent limitées pour traiter des échantillons de points de taille très importante par rapport à des approches plus locales.

Fusion d'images de profondeur La relation de voisinage 4-connexe des pixels des images de profondeur permet de définir des relations d'adjacence entre les points 3D correspondants. On peut ainsi représenter chaque image de profondeur comme une surface maillée. Si les images de profondeur correspondant aux différents points de vue d'acquisition se recouvrent suffisamment, il est alors possible d'exploiter leur représentation surfacique comme modèle de base pour reconstruire une surface qui approximera globalement la surface originale.

Curless et Levoy sont à l'origine d'une des premières méthodes de reconstruction de surface implicite à partir d'un ensemble d'images de profondeur [67]. La méthode définit la surface localement par une somme pondérée de fonctions de distance signées aux images de profondeur. Le champ de potentiel correspondant est échantillonné sur une grille régulière, construit de manière incrémentale en considérant une image de profondeur à la fois. A l'issue de ce processus, une surface triangulée peut être obtenue en appliquant à chaque cellule de la grille les règles de l'algorithme des Marching Cubes [133].

Cette méthode a été utilisée avec succès dans divers projets de numérisation d'œuvres d'art, comme le *Digital Michelangelo Project* [127]. Elle permet de traiter facilement des modèles comportant des centaines d'images de profondeur de plusieurs centaines de milliers de points. Rocchini et al. [164] ont proposé une technique similaire évitant le maintien d'une grille globale en ne considérant que les points d'intersection des images de profondeur avec les arêtes de la grille.

Estimation de la fonction caractéristique Étant donné un ensemble de points munis de normales orientées échantillonné à partir d'une surface fermée, Kazhdan [120] reconstruit une surface à partir d'une estimation de la fonction caractéristique du solide défini par l'échantillon de points, i.e. la fonction

qui vaut *un* à l'intérieur du solide et *zéro* à l'extérieur. L'originalité de la méthode est de considérer une représentation fréquentielle de cette fonction. Elle est estimée en calculant sa transformée de Fourier, dont les coefficients sont exprimés comme des intégrales de surface définies à partir des points de l'échantillon et de leurs normales en appliquant le *Théorème de la Divergence*. Ces intégrales de surface sont approximées par des sommes discrètes sur tous les points de l'échantillon (intégration de Monte-Carlo). Une fois les coefficients calculés, la fonction caractéristique est obtenue en leur appliquant la transformée de Fourier inverse. La surface reconstruite est l'isosurface correspondant à la valeur moyenne de la fonction caractéristique aux points de l'échantillon.

Pour réduire la complexité du calcul des coefficients de la transformée de Fourier, la première étape de l'algorithme consiste à construire un champ de potentiel dans un voisinage tubulaire autour de la surface à reconstruire par diffusion volumétrique en balayant une grille de voxels avec un filtre gaussien. Les coefficients de Fourier sont calculés par l'algorithme de transformée de Fourier rapide (*Fast Fourier Transform*), appliqué comme un filtre de convolution sur le champ de potentiel, centré sur les points de l'échantillon et aligné avec les normales. L'approximation des intégrales nécessite en principe que l'échantillon de points en entrée soit uniforme. En pratique, des échantillons de densité variable peuvent être traités en pondérant chaque point en fonction de la proximité de ses voisins, information qui peut être obtenue à l'issue de la phase de diffusion volumétrique. Les performances dépendent de la bande passante retenue pour reconstruire un modèle, qui détermine les fréquences les plus élevées prises en compte dans le calcul des coefficients de Fourier. Le niveau de détail de la reconstruction peut ainsi être facilement réglé en modifiant la valeur de la bande passante. Plus la bande passante est élevée, plus la surface sera représentée de manière détaillée, mais aussi coûteuse à calculer, nécessitant une grille de résolution de plus en plus fine $((2b)^3$ pour une bande passante b).

Estimation d'une fonction de distance non signée La construction de fonctions de distance signées nécessite que les points soient munis de normales orientées de façon consistante. Si celles-ci ne sont pas fournies, l'estimation d'orientations consistantes pose de nombreux problèmes. Les méthodes qui reposent sur des fonctions de distance signées posent souvent des problèmes de bruit topologique, i.e. la surface reconstruite n'a pas le même genre que la surface originale. Cela pose également des difficultés pour les algorithmes de maillage.

Hornung et Kobbelt [113] ont récemment proposé une méthode volumétrique de reconstruction de surface à partir d'échantillons de points non structurés qui repose sur une fonction de distance non signée à l'ensemble de points, échantillonnée sur une grille 3D. Les auteurs identifient les voxels traversés par la surface à partir d'une structure de graphe définie sur le champ de potentiel en mettant en œuvre un procédé de minimisation d'énergie combinatoire pour localiser la surface.

L'algorithme débute par une phase de diffusion volumétrique dans la grille permettant de construire un champ de distances non signées dans un voisinage tubulaire autour de l'échantillon de points. La diffusion est opérée en balayant la grille avec un filtre moyennneur, en ayant préalablement initialisé à *un* les voxels contenant des points et les autres à *zéro*. Un graphe étiqueté est ensuite construit sur les

voxels compris dans ce voisinage. Chaque sommet de ce graphe est associé à une face d'un voxel et les arêtes connectent les faces voisines au sens 8-connexe, avec pour chacune un poids qui dépend de la valeur de potentiel du voxel traversé. Un algorithme de calcul de coupe minimale permet de déterminer un ensemble optimal de voxels traversés par une approximation lisse de la surface originale.

Cette méthode, avec les RBF à support global, est certainement l'une des plus robustes actuellement pour la reconstruction de surface selon une approche implicite. Son principal inconvénient est sa dépendance par rapport à la résolution d'une grille.

1.1.2.2 Modèles déformables

Les méthodes de cette catégorie déforment une surface initiale jusqu'à obtenir une bonne approximation de l'échantillon de points. La surface est considérée comme une membrane élastique à laquelle est appliquée un processus de déformation pour minimiser une fonctionnelle d'énergie. Les modèles déformables paramétriques ont longtemps été étudiés, mais ceux-ci sont généralement restrictifs à la fois géométriquement et topologiquement. Les modèles implicites permettent de s'affranchir de ces restrictions. De manière générale, les méthodes s'appuyant sur des modèles déformables sont intéressantes pour leur robustesse, mais leur convergence n'est garantie que vers un minimum local, qui peut différer de la vraie surface. Nous présentons ici deux méthodes récentes utilisant un modèle de surface implicite.

Reconstruction par la méthode des level sets Zhao et al. [198] ont proposé une méthode de reconstruction de surface à partir d'un échantillon de points non organisé dont le principe est de contracter une surface fermée englobant l'ensemble de points. La surface est représentée implicitement selon la formulation dynamique des level sets.

Plus formellement, une surface \tilde{S} qui englobe l'ensemble de points est déformée de manière à minimiser une fonctionnelle d'énergie définie comme une aire pondérée, où chaque élément de surface est pondéré par sa distance au point de l'échantillon le plus proche.

$$E(\tilde{S}) = \left(\int_{\mathbf{x} \in \tilde{S}} d^p(\mathbf{x}) ds \right)^{1/p}, 1 \leq p \leq \infty$$

où $d(\mathbf{x})$ est la distance entre $\mathbf{x} \in \mathbb{R}^3$ et le point le plus proche dans P , et ds un élément de surface. La fonctionnelle d'énergie E est minimisée par descente de gradient jusqu'à un minimum local en contractant la surface englobante initiale. Tout point \mathbf{x} de \tilde{S} à un instant donné se déplace vers l'intérieur de la surface dans la direction de sa normale $\mathbf{n}(\mathbf{x})$ avec une vitesse proportionnelle à $-\nabla d(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}) + d(\mathbf{x})\kappa$, où κ désigne la courbure moyenne. Le terme $-\nabla d(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x})$ traduit l'attraction de la surface \tilde{S} vers les données, tandis que $d(\mathbf{x})\kappa$ représente sa tension. La surface évolue jusqu'à atteindre un état d'équilibre, caractérisé par l'équation :

$$\nabla d(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}) + d(\mathbf{x})\kappa = 0$$

Pour initialiser la surface \tilde{S} , Zhao et al. calculent une approximation initiale de S en prenant uniquement

en compte le champ de gradient $-\nabla d$; ce schéma d'initialisation revient à effectuer la convection d'une surface englobante dans $-\nabla d$. Par abus de langage, on parle de *modèle de convection*. À condition que l'initialisation soit proche de l'optimum, cette méthode est rapide et robuste au bruit. Au cours de son évolution, la surface peut également changer facilement de topologie, ce qui constitue un bénéfice important de la représentation implicite.

Évolution d'une surface triangulée guidée par un champ de potentiel Sharf et al. [175] ont récemment proposé une méthode reposant sur un modèle de surface déformable explicite, représentée par un maillage, évoluant à travers un champ de potentiel. Une surface triangulée initiale incluse dans le volume de l'objet est déformée de manière locale en déplaçant ses sommets dans la direction de leur normale. L'évolution de cette surface est guidée par un champ de distances signées échantillonné sur une grille adaptative qui définit une représentation implicite de la surface cible. Au cours de ce processus, le déplacement des sommets du maillage est contraint pour maintenir la cohérence de la surface qui évolue (système laplacien). Les auteurs contrôlent le niveau de détail de la reconstruction en considérant plusieurs fronts d'évolution indépendants avec des propriétés différentes d'adaptation à la forme locale. Plusieurs fronts peuvent fusionner, donnant lieu à des changements topologiques. Le maillage est optimisé de manière dynamique en fonction de la géométrie locale, et le processus s'achève par une phase de projection MLS pour capturer les détails fins.

1.2 Simplification d'échantillons de points

Pour représenter la surface d'objets complexes, les échantillons issus de leur numérisation peuvent contenir un très grand nombre de points. Les techniques de simplification d'échantillons de points analysent la pertinence des données dans le but de fournir des modèles comportant moins de points que les modèles de départ, tout en représentant une surface proche de l'originale.

Le problème de la simplification d'échantillons de points peut être formalisé comme suit. Étant donné un ensemble de points P échantillonnant une surface S , trouver un échantillon de points P' avec $|P'| < |P|$ tel que P' échantillonne une surface S' proche de S . Cet objectif nécessite de définir une mesure d'erreur géométrique entre la surface originale et la surface simplifiée, pour laquelle les méthodes vont recourir à des estimations de propriétés globales ou locales de la surface originale, et un schéma de simplification. Comme pour la simplification de surfaces maillées, trouver l'optimum global de ce problème est NP-difficile [98], c'est pourquoi les méthodes mettent en œuvres diverses heuristiques pour guider le processus de simplification. Il est également important de contrôler la densité et la distribution des points de l'échantillon simplifié de manière à disposer de conditions d'échantillonnage favorables pour produire une surface maillée par exemple.

Il existe deux catégories principales d'algorithmes pour simplifier un échantillon de points : les algorithmes de sous-échantillonnage et les algorithmes de rééchantillonnage. Les algorithmes de sous-échantillonnage produisent un échantillon de points simplifié qui est un sous-ensemble du nuage de

points original, alors que les algorithmes de rééchantillonnage reposent sur l'estimation de propriétés de la surface échantillonnée pour calculer de nouveaux points pertinents. Dans la littérature, ces principes ont été appliqués selon trois schémas de simplification principaux, que nous allons examiner en détails : la simplification par sélection ou calcul de points représentants de sous-ensembles de l'échantillon initial, la simplification itérative, et la simplification par échantillonnage incrémental.

1.2.1 Sélection ou calcul de points représentants de sous-ensembles

Un première stratégie pour simplifier un échantillon de points consiste à le décomposer en petites régions dont chacune est représentée par un seul point dans l'échantillon simplifié. Les méthodes de cette catégorie se distinguent par les critères définissant les régions et leur construction.

Pauly et al. [160] ont proposé une méthode reposant sur une décomposition hiérarchique de l'échantillon de points, calculée par partition binaire de l'espace. Les plans de coupe sont définis par le centre et la direction principale de chaque région. Le critère de partitionnement dépend à la fois d'un nombre maximum de points et des variations de la géométrie locale dans une région. Les auteurs proposent de mesurer ces variations à partir des trois valeurs propres $\lambda_1 \leq \lambda_2 \leq \lambda_3$ de la matrice de covariance associée, en considérant le rapport $\frac{\lambda_1}{\lambda_1 + \lambda_2 + \lambda_3}$. Ce rapport mesure l'anisotropie locale de la distribution de l'échantillon, reflétant la courbure locale. De par la nature spatiale de cette approche, il est difficile de contrôler la qualité de la distribution des points sur la surface échantillonnée.

Une autre méthode proposée par Pauly et al. [160] consiste à décomposer l'échantillon de points par croissance de régions. À partir d'un point choisi de manière aléatoire, une région est construite en ajoutant successivement ses plus proches voisins tant qu'une limite de taille n'est pas atteinte, et/ou qu'un seuil de variation géométrique calculé selon la mesure précédente n'est pas atteint par les points de cette région. Les régions suivantes sont construites à partir de points voisins des régions déjà créées, en excluant du processus les points de ces dernières. Un inconvénient de cette méthode est qu'elle crée de nombreuses régions qui n'atteignent pas une taille maximale. Cette fragmentation résulte en une mauvaise distribution des points dans l'échantillon simplifié. Pour y remédier, les auteurs proposent d'éliminer ces trop petites régions en distribuant leurs points dans les régions voisines de taille suffisante, ce qui peut conduire à dépasser les bornes fixées au départ.

D'autres méthodes adoptant le principe de création locale de régions reposent sur un principe de couverture de l'échantillon de points, permettant de prévenir la fragmentation. Avec pour objectif d'accélérer le rendu de surfaces représentées par des points, Wu et Kobbelt [190] calculent un ensemble optimal de splats pour couvrir une surface échantillonnée. La première étape de la méthode consiste à approximer localement la surface en chaque point de l'échantillon par un élément de surface circulaire ou elliptique plan appelé *splat*, dont la taille (rayon ou dimensions des deux axes principaux) est adaptée à la courbure locale de la surface. Chaque splat est orienté selon la normale au point associé, et sa taille déterminée par un processus de croissance de région en considérant la distance entre son plan et les points du voisinage. Dans un second temps, les splats redondants sont éliminés au cours d'un processus de filtrage de type expansion de surface. Pour garantir le recouvrement de la totalité de la surface échantillonnée, l'algo-

rithme procède comme suit. Pour chaque splat traité, les points qu'il recouvre sont projetés sur son plan, puis seuls les splats associés aux points qui se projettent à l'intérieur de l'enveloppe convexe des points projetés sont éliminés. Au cours de ce processus, la régularité de la distribution n'est pas contrôlée. Une phase de relaxation peut être appliquée pour déterminer une position optimale pour les splats restants. Cette méthode permet de générer des couvertures de splats de grande qualité pour des surfaces lisses, en filtrant le bruit. Cependant, cette méthode est pénalisée par la coût de son initialisation et celui de la phase de relaxation pour des échantillons de points de taille importante.

Les limites de la méthode précédente ont été dépassées par Ohtake et al. [156], dont le but de générer un maillage simplifié directement à partir d'un échantillon de points. Pour cela, un nouvel échantillon est construit à partir d'une couverture de sphères, selon une stratégie de minimisation d'erreur quadratique locale. Une surface triangulée duale de la couverture de sphères est ensuite construite par expansion de surface. Nous allons décrire en détails la première étape de la méthode qui consiste à construire une couverture de sphères dont le rayon s'adapte à la courbure locale, avec pour chacune d'elle le calcul d'un point représentatif du voisinage de points qu'elle contient. Chaque sphère est associée à une fonction d'erreur quadratique de la forme :

$$Q(\mathbf{c}, r, \mathbf{x}) = \sum_j G_{Tr}(\|\mathbf{p}_j - \mathbf{c}\|)(\mathbf{n}_j \cdot (\mathbf{x} - \mathbf{p}_j))^2$$

où \mathbf{c} est le centre de la sphère, r son rayon, \mathbf{x} un point de l'espace, et G_{Tr} une fonction de poids monotone décroissante. La fonction Q correspond à une somme pondérée des distances au carré du point \mathbf{x} aux plans tangents aux points \mathbf{p}_j contenus dans une boule de centre \mathbf{c} et de rayon d'influence Tr (en général $T = 2$). Le premier objectif est de trouver le point $\mathbf{x}_{\min}(r)$ qui minimise l'erreur

$$E(r) = \sqrt{Q(\mathbf{c}, r_{err}, \mathbf{x}_{\min}(r))}$$

pour un rayon r initial fixé. Cette erreur correspond à la capacité du point $\mathbf{x}_{\min}(r)$ à représenter géométriquement les points contenus dans la boule de rayon r . Ensuite, le rayon r est raffiné par dichotomie de manière à ce que $E(r) \approx T_{err}$ où T_{err} est une tolérance d'erreur fixée. La couverture de sphère est construite de manière itérative selon un parcours en largeur des points de l'échantillon, en prenant à chaque fois un point de l'échantillon comme centre d'une nouvelle sphère. Pour chaque sphère traitée, les points qu'elle contient sont projetés sur le plan tangent en son centre, puis ceux qui retrouvent à l'intérieurs à l'enveloppe convexe de ces points sont éliminés, principe similaire à celui de Wu et Kobbelt [190] pour l'élimination des splats redondants.

La technique de rééchantillonnage mise en œuvre par Ohtake et al. [156] permet de générer très efficacement un échantillon dont la densité s'adapte à la géométrie locale, avec de bonnes propriétés de distribution. En outre, la mesure d'erreur considérée permet de détecter et de restaurer les arêtes vives de surfaces non lisses. Une procédure de triangulation locale ad hoc permet également de construire un maillage rapidement à partir de la couverture de sphères. Cette méthode reste cependant fortement dépendante de la qualité des normales et peut échouer dans les régions où deux nappes de la surface

échantillonnée sont proches.

1.2.2 Simplification itérative

Le principe de la simplification itérative est de supprimer des points de l'échantillon initial de manière incrémentale en fonction de critères géométriques mesurant localement la redondance des données.

L'un des premiers algorithmes de simplification d'échantillons de points a été proposé par Linsen [130], qui associe une valeur scalaire à chaque point mesurant localement la variation moyenne de certaines informations, telles que la proximité des voisins ou la direction des normales. Les points ayant la mesure la plus faible sont retirés de manière itérative. L'algorithme présente cependant l'inconvénient de ne donner aucune garantie sur la densité de l'ensemble de points résultant.

Dey et al. [77] utilisent une approximation du LFS de la surface échantillonnée. Cette approximation est calculée à partir de la triangulation de Delaunay de l'échantillon de points en entrée, ce qui représente un inconvénient pour des échantillons de taille très importante. En chaque point \mathbf{p} de l'échantillon, le rapport entre la valeur estimée du LFS et le rayon de l'intersection de la cellule de Voronoï de \mathbf{p} avec le *cocône* [14] en \mathbf{p} fournit une mesure d'erreur. Les points en dessous d'un certain seuil d'erreur sont éliminés. Cette méthode ne contrôle pas la distribution des points dans l'échantillon simplifié. Par la suite, Dey et al. ont étendu leur méthode en détectant localement les régions qui deviennent sous-échantillonnées relativement à la mesure d'erreur précédente pour y ajouter des points par rééchantillonnage [79]. Les points ajoutés correspondent à des points d'intersection entre les cocônes et les arêtes de Voronoï les intersectant, donc situés à équidistance de points existants, ce qui permet de maintenir une bonne distribution.

Alexa et al. [9] estiment les propriétés géométriques locales de la surface échantillonnée en utilisant un modèle Moving Least Squares de la surface sous-jacente, ce qui nécessite de disposer de normales orientées de façon consistante. Ils calculent la contribution d'un point à cette surface en le projetant sur une surface MLS estimée à partir des points voisins. La distance entre la position du point et sa projection sur la surface fournit une mesure d'erreur. Les points pour lesquels cette distance est la moins importante sont retirés. Cette méthode ne garantit pas la densité de l'échantillon de points résultant. Pour compenser, Alexa et al. [9] proposent d'enrichir l'échantillon dans les régions sous-échantillonnées en considérant la projection de celles-ci sur un plan. Ils calculent le diagramme de Voronoï plan des points projetés de manière à insérer de nouveaux points à équidistance des premiers. Ces nouveaux points sont ensuite remontés sur la surface en utilisant l'opérateur de projection. Le processus est répété jusqu'à ce que la distance euclidienne entre le point suivant à ajouter et le point existant le plus proche devienne inférieure à un certain seuil. Si cette méthode permet d'obtenir des résultats de qualité, l'utilisation intensive de l'opérateur de projection MLS la rend coûteuse pour des échantillons de taille très importante.

Pauly et al. [160] ont étendu directement la technique de simplification de maillage de Garland et Heckbert [99] aux échantillons de points en considérant les relations de plus proches voisins comme relations de connectivité. Des paires de plus proches voisins sont ainsi contractées, remplaçant deux

points par un nouveau point calculé comme une moyenne pondérée des premiers. Le coût de chaque opération de contraction est mesuré en adaptant la mesure d'erreur proposée par Garland et Heckbert, dont l'idée est d'approximer la surface localement par un ensemble de plans tangents et d'estimer la déviation géométrique d'un point par rapport à la surface représentée par la somme des distances au carré à ces plans. L'erreur quadratique ainsi définie pour chaque point \mathbf{p} est initialisée en estimant un plan tangent pour chaque arête qui connecte \mathbf{p} avec ses plus proches voisins. Cette méthode offre l'avantage de contrôler la distribution de l'échantillon simplifié, qui possède par ailleurs la propriété de préserver les détails. Cependant, son coût d'initialisation est élevé et elle nécessite le maintien d'une file de priorité globale, ce qui représente un inconvénient pour de gros échantillons de points.

Il est intéressant de noter que cette dernière méthode et celle d'Ohtake et al. [156] présentée plus haut sont les seules à permettre un contrôle du niveau de détail de la forme de la surface représentée par l'échantillon de points simplifié. L'évolution de la forme de la surface sous-jacente nécessite un rééchantillonnage. Dans les autres cas, la mesure d'erreur est une mesure de précision dans le sens où elle contrôle uniquement la redondance des données. Cela sera aussi le cas dans le cadre de notre approche dynamique.

1.2.3 Simplification par échantillonnage incrémental

À l'inverse de la simplification itérative, l'échantillon de points simplifié peut être construit par enrichissement progressif d'un sous-ensemble de points initial, ou échantillonnage d'une surface implicite.

Boissonnat et Cazals [40] ont proposé une technique de reconstruction de surface qui enrichit progressivement un sous-ensemble de points de départ choisis de manière aléatoire. À partir de la triangulation de Delaunay de cet ensemble est estimée une fonction de distance signée à la surface échantillonnée par interpolation des voisins naturels de fonctions de distances signées définies à partir des plans tangents en chaque point [39]. Cette fonction est utilisée pour enrichir l'ensemble de points initial jusqu'à ce qu'un nombre suffisant de points se retrouve en dessous d'une certaine tolérance d'erreur. Une surface est ensuite reconstruite à partir de la triangulation de Delaunay de l'échantillon de points enrichi. Si cette surface ne vérifie pas une condition d'erreur, des points supplémentaires peuvent être insérés itérativement. Une idée intéressante est que la triangulation de Delaunay n'est calculée que pour les points conservés. Cependant, des files de priorité nécessitent d'être mises à jour pour maintenir la représentation implicite et des normales orientées si elles ne sont pas fournies ou préalablement estimées, ce qui représente un surcoût important. La régularité de la distribution de l'échantillon simplifié n'est également pas contrôlée dans cette méthode.

Une autre technique consiste à considérer une représentation implicite de la surface échantillonnée dans sa globalité et d'appliquer un schéma d'échantillonnage de type *Farthest Point Sampling*, tel que nommé en traitement d'image. Il s'agit d'une technique itérative dont le principe est de choisir le point suivant dans la zone la moins échantillonnée, ce qui permet d'obtenir une distribution de l'échantillon uniforme localement. Ce principe était déjà mis en œuvre par Dey et al. [79] et Alexa et al. [9] dans les phases d'enrichissement de leurs méthodes de simplification itérative.

Moening et Dogson [144] ont développé une méthode reposant sur une représentation par un champ de distance échantillonné sur une grille régulière. Le principe est de sélectionner les points en construisant un diagramme de Voronoï géodésique de manière incrémentale selon un principe de propagation de fronts. À chaque instant, le point suivant est choisi dans la zone la moins échantillonnée de la surface. Dans [145], les mêmes auteurs ont étendu leur technique de sous-échantillonnage au rééchantillonnage en prenant comme points les intersections des arêtes du diagramme de Voronoï avec la surface. Un inconvénient de cette méthode est sa dépendance vis à vis de la résolution de la grille utilisée, qui joue sur la qualité du résultat en dépit de la précision des données.

Boissonnat et Oudot [42, 43, 44] ont récemment revisité la technique d'échantillonnage de surface de Chew [57] pour générer des triangulations optimales pour une surface implicite ou polyédrique fixée. L'algorithme de Chew maintient la triangulation de Delaunay restreinte d'un échantillon de points généré de manière incrémentale sur une surface lisse. Après convergence, les propriétés de l'échantillon généré par l'algorithme de Chew permettent de faire de cet ensemble de facettes une bonne approximation de la surface, avec des garanties à la fois géométriques et topologiques. Plus précisément, Boissonnat et Oudot [44] ont prouvé que pour toute surface fermée S et tout paramètre $\varepsilon > 0$, l'algorithme de Chew produit un ε -échantillon lâche P de S tel que les angles internes des facettes de la triangulation de Delaunay de P restreinte à S est au moins $\frac{\pi}{6}$. Si S est une surface lisse dont les normales vérifient une condition de Lipschitz, alors pour un ε suffisamment petit, la triangulation de Delaunay de P restreinte à S forme une bonne approximation de S géométriquement et topologiquement. Suivant cette approche, Boissonnat et Oudot reconstruisent une surface à partir d'un ensemble de points en s'appuyant sur une représentation implicite de la surface par Moving Least Squares selon le modèle proposé par Adamson et al. [3]. Un inconvénient de la méthode est la nécessité d'estimer le LFS en des points sur la surface, ce qui nécessite de construire une triangulation de Delaunay des points de l'échantillon en entrée.

Pauly et Gross [159] ont proposé une technique de rééchantillonnage fondée sur la théorie de Fourier. L'échantillon de point est tout d'abord décomposé en sous-ensembles projetables sur un plan. Chaque sous-ensemble est alors converti en une image de profondeur. Une décomposition spectrale permet ensuite de générer un échantillon dont la densité s'adapte localement à la fréquence du signal géométrique. Pour éviter des discontinuités d'échantillonnage et dans la direction des normales aux bords des patches obtenus, ceux-ci sont raccordés de manière lisse en calculant une combinaison convexe des points situés près des bords, pondérés par une fonction de mélange. Un inconvénient de la méthode est sa dépendance vis à vis de la décomposition initiale de l'échantillon de points.

Pauly et al. [160] ont également développé une méthode construisant un système de particules sur une surface MLS selon le principe de la méthode de Witkin et Heckbert [189], dans le but de produire un échantillon de point simplifié. Du fait de la nature globale du processus de relaxation qu'elle implique, cette technique est coûteuse et peut être adaptée à de gros échantillons de points.

1.3 Synthèse

Dans ce chapitre, nous avons présenté les principales techniques existantes de reconstruction de surface et de simplification d'échantillons de points.

Les méthodes de reconstruction combinatoires construisent généralement une surface triangulée interpolant les points de l'échantillon en entrée. Les méthodes s'appuyant sur la triangulation de Delaunay sont les plus générales, pour la plupart peu contraignantes, et pour certaines, capables d'apporter des garanties sous de bonnes conditions d'échantillonnage. Toutes les informations nécessaires aux algorithmes peuvent être tirées de cette seule structure géométrique. Le principal inconvénient de ces méthodes réside dans la construction et le maintien d'une structure de données impliquant l'ensemble des points et sensible à la position de chacun d'entre eux. L'adaptation de ces algorithmes à des échantillons de taille importante est par conséquent difficile. Les techniques plus locales sont moins contraignantes en ce sens, mais elles sont globalement moins robustes.

Les approches par ajustement de modèles font davantage d'hypothèses sur la forme de la surface à reconstruire. Elles sont particulièrement intéressantes pour leur pouvoir d'approximation et leur capacité à reconstituer la surface dans les régions où il manque des données. Certaines permettent également de contrôler le niveau de détail de la reconstruction. Les techniques procédant par mélange d'approximations locales, comme celle d'Ohtake et al. [152] ou le modèle de surface MLS [126], offrent une souplesse particulière dans la mesure où la surface peut être mise à jour facilement si les données sont modifiées. Ces méthodes sont cependant relativement sensibles à la qualité des données en entrée, en particulier des normales. Si celle-ci est insuffisante, la fonction définissant la surface peut comporter des artefacts sans rapport avec la surface échantillonnée. Une contrainte importante des méthodes procédant par mélange d'approximations locales ou la définition de points de contraintes (cas des RBF) est la nécessité de disposer de normales orientées de manière cohérente sur l'échantillon de points. Une autre contrainte est la nécessité de rééchantillonner la surface, ou de la mailler pour visualiser la reconstruction interactivement.

Malgré la distinction que nous avons opérée entre l'approche combinatoire et l'approche modèle, il ressort une forte tendance au développement de méthodes hybrides, traduisant le rapprochement récent des communautés de l'informatique graphique et de la géométrie algorithmique. En particulier, des méthodes combinatoires globales contribuent à améliorer la robustesse de méthodes reposant sur des modèles locaux. Par exemple Boissonnat et Cazals [39] utilisent les voisins naturels, définis à partir du diagramme de Voronoï de l'échantillon de points, pour combiner des fonctions de distance à des plans tangents. Samozino et al. [165] utilisent également le diagramme de Voronoï d'un échantillon de points pour apporter davantage de robustesse à la reconstruction par RBF à support compact. Hornung et Kobbelt [113] minimisent une énergie combinatoire sur un champ de distances discret pour modéliser une surface implicite ayant de bonnes propriétés.

Dans cette thèse, nous explorons une autre manière d'associer les deux types d'approches. Notre objectif est de bénéficier des avantages de l'approche combinatoire globale de la reconstruction de surface

tout en y apportant de la flexibilité par le recours à des modèles locaux pour analyser la pertinence des données en entrée et éliminer les données redondantes.

Dans le cas de données redondantes, il est souhaitable de faire l'économie d'une triangulation de Delaunay globale en simplifiant les données. Les méthodes de simplification d'échantillons de points s'appuyant sur des modèles locaux sont les plus recommandées pour atteindre cet objectif. Bien que simplification d'échantillon et reconstruction de surface soient souvent abordées de manière indépendantes, la simplification met généralement en œuvre des mécanismes proches de la reconstruction de surface. Quelques méthodes se sont cependant intéressées à la manière de les faire coopérer. Boissonnat et Cazals [40] ont abordé ce thème sous l'angle de l'enrichissement progressif d'un sous-échantillon initial, mais ils ne contrôlent pas la qualité de la distribution de l'échantillon. Ohtake et al. [156] exploitent les propriétés d'une couverture de sphères pour construire un maillage, mais cette technique purement locale manque de robustesse. Nous proposons une approche différente reposant sur l'algorithme de convection géométrique évoqué précédemment, dans laquelle la reconstruction guide la simplification avec un contrôle de la densité et de la distribution de l'échantillon.

Enfin, si certaines méthodes peuvent reconstruire directement des surfaces simplifiées, elles ne permettent pas de mettre à jour facilement le résultat d'une reconstruction. La méthode de Boissonnat et Cazals [40] permet d'enrichir l'échantillon de points simplifié, mais elle présente l'inconvénient de nécessiter des normales orientées, et n'a pas été étendue au retrait de données. Pourtant, la capacité de mise à jour peut être très intéressante dans le cadre d'un projet de numérisation où toutes les données ne sont pas disponibles au même moment, pour reconstruire un modèle progressivement lorsque des données sont transmises sur un réseau, ou encore pour pouvoir régler différents paramètres sans reprendre le processus de reconstruction à zéro. A notre connaissance, aucune méthode de reconstruction de surface combinatoire ne s'est directement intéressée à la possibilité de mettre à jour le résultat de la reconstruction a posteriori en ajoutant ou en retirant des données dynamiquement. Nous proposons de combler cette lacune en rendant dynamique notre approche de la reconstruction.

Reconstruction de surface par convection géométrique

Dans ce chapitre, nous décrivons en détail l’algorithme de reconstruction de surface par convection géométrique introduit par Chaine [52]. Rappelons qu’il s’agit d’un algorithme de nature combinatoire fondé sur la triangulation de Delaunay, qui entre dans la classe des méthodes volumiques. Dans les paragraphes suivants, nous rappelons le principe du schéma de convection, pour en arriver à son interprétation géométrique (section 2.1). Nous présentons ensuite l’algorithme original de convection géométrique (section 2.2). Nous en proposons une lecture en termes de prédicats et de requêtes, dont nous ferons usage pour décrire notre algorithme de reconstruction sélective. Enfin, nous examinons quelques propriétés de l’algorithme pertinentes pour notre approche dynamique (section 2.3).

2.1 Schéma de convection et interprétation géométrique

L’algorithme de convection géométrique s’appuie sur le schéma de convection introduit par Zhao, Osher et Fedkiw dans leur méthode de reconstruction de surface fondée sur les level sets [198]. Zhao et al. utilisent ce schéma pour calculer une première surface, proche de la surface optimale qu’ils cherchent à obtenir en minimisant une fonctionnelle d’énergie. L’algorithme de convection géométrique permet de calculer efficacement le résultat de la convection en tirant parti de résultats de géométrie algorithmique.

Le schéma de convection consiste à effectuer la convection d’une surface flexible \hat{S} englobant l’échantillon de points dans le champ de gradient de la fonction de distance au point de l’échantillon le plus proche définie par $d(\mathbf{x}) = \min_{\mathbf{p} \in P} \|\mathbf{x} - \mathbf{p}\|$ pour tout $\mathbf{x} \in \hat{S}$. Les points de la surface se déplacent dans la direction de leur normale $\mathbf{n}(\mathbf{x})$ avec une vitesse proportionnelle à $-\mathbf{n}(\mathbf{x}) \cdot \nabla d(\mathbf{x})$. Le processus s’arrête lorsque la surface atteint un état stationnaire, caractérisé par l’équation $-\mathbf{n}(\mathbf{x}) \cdot \nabla d(\mathbf{x}) = 0$.

La figure 2.1 illustre le schéma de convection sur un échantillon de points 2D, en mettant en évidence la relation qui existe avec son diagramme de Voronoï : tous les points qui appartiennent à un morceau de la surface situés dans une même cellule de Voronoï sont attirés vers un même point de l’échantillon. Chaine [52] a prouvé le théorème suivant.

Théorème 1. *Étant donné une surface fermée \hat{S} englobant un ensemble de points P , la convection de \hat{S} à travers le champ de gradient $-\nabla d$ converge vers une pseudo-surface linéaire par morceaux fermée plongée dans la triangulation de Delaunay de P telle que :*

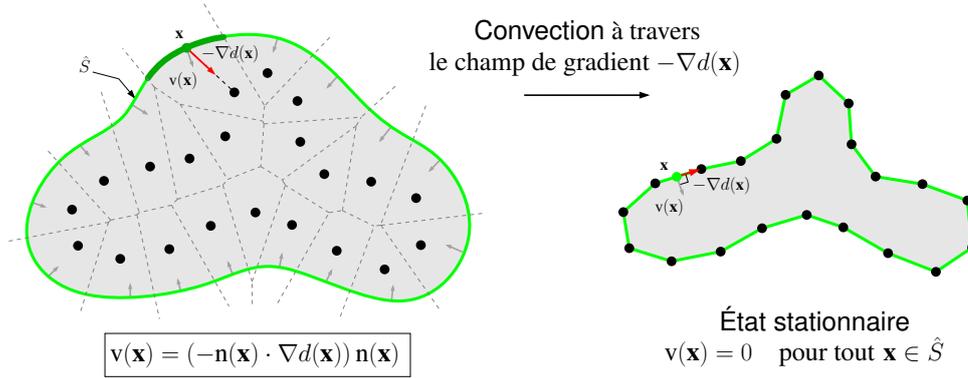


FIG. 2.1 – Schéma de convection en 2D.

- toutes ses facettes sont orientées vers l'intérieur ;
- pour chaque facette, la demi-sphère circonscrite diamétrale orientée vers l'intérieur de la surface est vide de points de P (propriété de Gabriel orientée).

La propriété de Gabriel orientée découle de l'interaction de la surface en convection avec le diagramme de Voronoï de l'échantillon de points. Lorsqu'une facette \mathbf{pqr} ne vérifie pas la propriété de Gabriel orientée, cela implique que la surface intersecte la cellule de Voronoï d'un point \mathbf{s} qui devient un nouvel attracteur local de la surface \hat{S} . La figure 2.2 illustre en deux dimensions l'interprétation géométrique locale du schéma de convection.

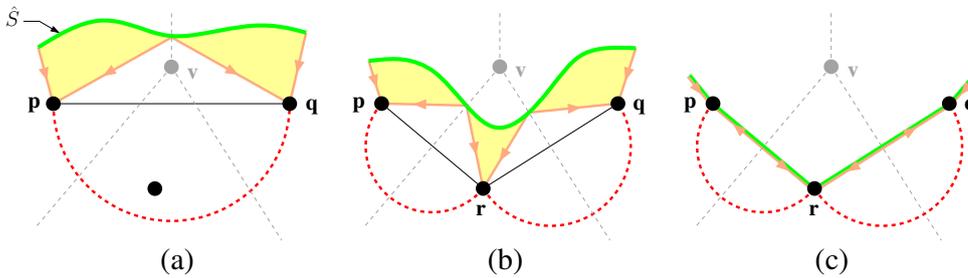


FIG. 2.2 – Interprétation géométrique locale du schéma de convection en 2D. En (a), La portion de courbe (en vert) est attirée par les points \mathbf{p} et \mathbf{q} , qui forment une arête de Delaunay avec une demi-sphère de Gabriel non-vide. Lorsque la courbe traverse le sommet de Voronoï \mathbf{v} , sa portion qui intersecte la cellule de Voronoï du point \mathbf{r} est attirée vers ce point.

Le terme *pseudo-surface* signifie que différentes parties de cette surface peuvent être pincées, c'est-à-dire qu'elles peuvent localement partager une géométrie commune tout en demeurant indépendantes topologiquement. Plus formellement, une pseudo-surface peut être définie comme le plongement géométrique d'une variété combinatoire orientable telle que les images géométriques de deux sommets non voisins, arêtes ou facettes peuvent être identiques, disjointes, ou s'intersecter sur leurs bords. Par la suite, nous y ferons simplement référence par le terme surface.

Zhao et al. [198] calculent le résultat de la convection dans un cadre implicite sur une grille régulière avec un algorithme d'étiquetage rapide (*fast tagging algorithm*). Chaîne a proposé un algorithme efficace

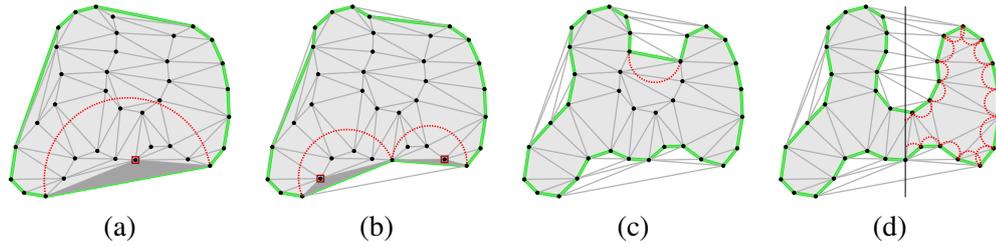


FIG. 2.3 – Convection géométrique vers un ensemble de points 2D. En (a), la courbe est initialisée sur l’enveloppe convexe de l’ensemble de points. L’arête courante, circonscrite par une demi-sphère de Gabriel non vide, forme un triangle de Delaunay (gris foncé) avec le point. Ce triangle devient externe, la courbe est mise à jour en (b), et celle-ci continue de se contracter. En (c), une arête bloquant une poche est détectée ; elle sera forcée. Le résultat final apparaît en (d) avec des demi-sphères de Gabriel vides.

permettant d’obtenir le résultat de la convection uniquement en considérant les données et en exploitant le théorème 1.

2.2 Analyse de l’algorithme : requêtes et prédicats mis en œuvre

Le théorème 1 donne lieu à un algorithme qui permet de réaliser le schéma de convection entièrement dans la triangulation de Delaunay de l’échantillon de points P , jusqu’à obtenir le résultat de la convergence dans le champ de gradient $-\nabla d$. L’algorithme consiste à retirer les facettes qui ne vérifient pas la propriété de Gabriel orientée selon un processus itératif de sculpture partant de l’enveloppe convexe de P . Il est aussi possible d’initialiser la surface à partir d’une boîte englobante en ajoutant ses coins aux données d’entrée. L’évolution de la surface \hat{S} au cours du processus de sculpture est guidée localement par un prédicat géométrique \mathbf{P}_{go} et une requête géométrique \mathbf{Q}_{dt} définis comme suit :

(\mathbf{P}_{go}) Étant donné une facette orientée \mathbf{pqr} , tester si elle satisfait la propriété de Gabriel orientée.

(\mathbf{Q}_{dt}) Étant donné une facette orientée \mathbf{pqr} , obtenir le point $\mathbf{s} \in P$ tel que \mathbf{pqrs} forme un tétraèdre de Delaunay dans le demi-espace situé au-dessus du plan de la facette.

Le demi-espace situé au-dessus du plan de la facette est celui qui fait face à l’intérieur de la surface.

L’algorithme de reconstruction de surface est illustré en deux dimensions par la figure 2.3, et résumé par l’algorithme 1. La surface \hat{S} est initialisée sur l’enveloppe convexe de P , toutes les facettes orientées vers l’intérieur. Celles-ci sont ensuite traitées indépendamment. L’ordre de traitement n’a pas d’importance si on ne s’intéresse qu’au résultat final ; un traitement *first in first out* des facettes introduites dans la surface est cependant plus proche de l’évolution de la surface dans le temps pendant le processus de convection, ce qui revient à effectuer un parcours en largeur des cellules de la triangulation de Delaunay externes à la surface. Si une facette \mathbf{pqr} ne vérifie pas le prédicat \mathbf{P}_{go} , celle-ci est *ouverte* vers le point \mathbf{s} formant une cellule de Delaunay avec \mathbf{pqr} , i.e. le point répondant à la requête \mathbf{Q}_{dt} . Cette opération revient à retirer la facette \mathbf{pqr} de \hat{S} pour la remplacer par les trois autres facettes, orientées de manière consistante, du tétraèdre de Delaunay auquel elle appartient. Il est à noter qu’une facette peut être ouverte vers un point déjà attaché à un sommet de la pseudo-surface courante. Deux facettes de la surface

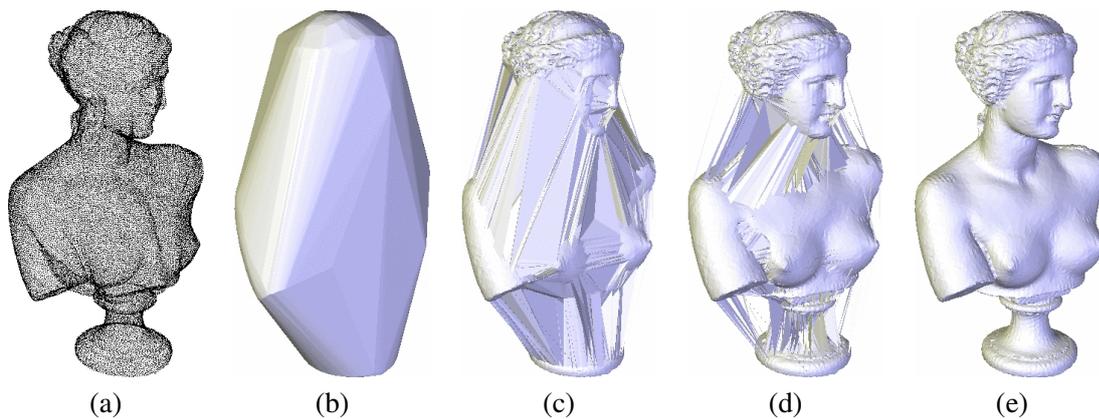


FIG. 2.4 – Convection géométrique sur un échantillon de points en 3D (modèle APHRODITE); (a) : échantillon de points ; (b) : initialisation de la surface sur l'enveloppe convexe ; (c) et (d) : deux étapes du processus d'évolution de surface ; (e) : résultat de l'algorithme après convergence.

partageant la même géométrie sont dites *couplées* ; elles ont nécessairement des orientations opposées. Au cours de la convection, il peut se produire un effondrement au niveau de deux facettes couplées dont une ne vérifie pas la propriété de Gabriel orientée. Celles-ci sont supprimées, ce qui peut entraîner un changement topologique de la pseudo-surface, comme la création d'une poignée ou d'une nouvelle composante connexe (fig. 2.5). Le processus de convection s'arrête dès que toutes les facettes de la surface vérifient le prédicat \mathbf{P}_{go} .

Il est nécessaire de souligner qu'en présence de cavités plus larges que les demi-sphères de Gabriel associées aux facettes de la surface, le processus de convection peut s'arrêter prématurément. Si l'échantillon de points est suffisamment dense, le processus de convection est poursuivi localement en réalisant une étude géométrique locale. Chaine propose de détecter les inconsistances entre la taille des facettes bloquantes et la densité locale. Une solution nécessitant la mise en œuvre d'une heuristique globale consiste à considérer un critère de persistance topologique [85, 101]. Il est à noter qu'en deux dimensions, les changements topologiques d'une courbe en convection correspondant à l'apparition de nouvelles composantes connexes ne peuvent se produire que par résolution de poches.

La surface résultante peut contenir des facettes orientées couplées, appelées *parties minces*. Certaines de ces parties minces n'appartiennent pas au résultat de la convection, aussi il est important d'identifier celles à conserver. Cette sélection est accomplie en poursuivant le processus de convection en 2D sur les parties minces, à partir de leur bord.

Complexité En supposant que la triangulation de Delaunay de l'échantillon de points a été construite, le prédicat \mathbf{P}_{go} et la requête \mathbf{Q}_{dt} sont tous deux évalués en temps constant, et la complexité de l'algorithme est linéaire par rapport au nombre de cellules de Delaunay traversées par la surface. Ce nombre dépend à la fois de la forme de la surface et de son échantillonnage.

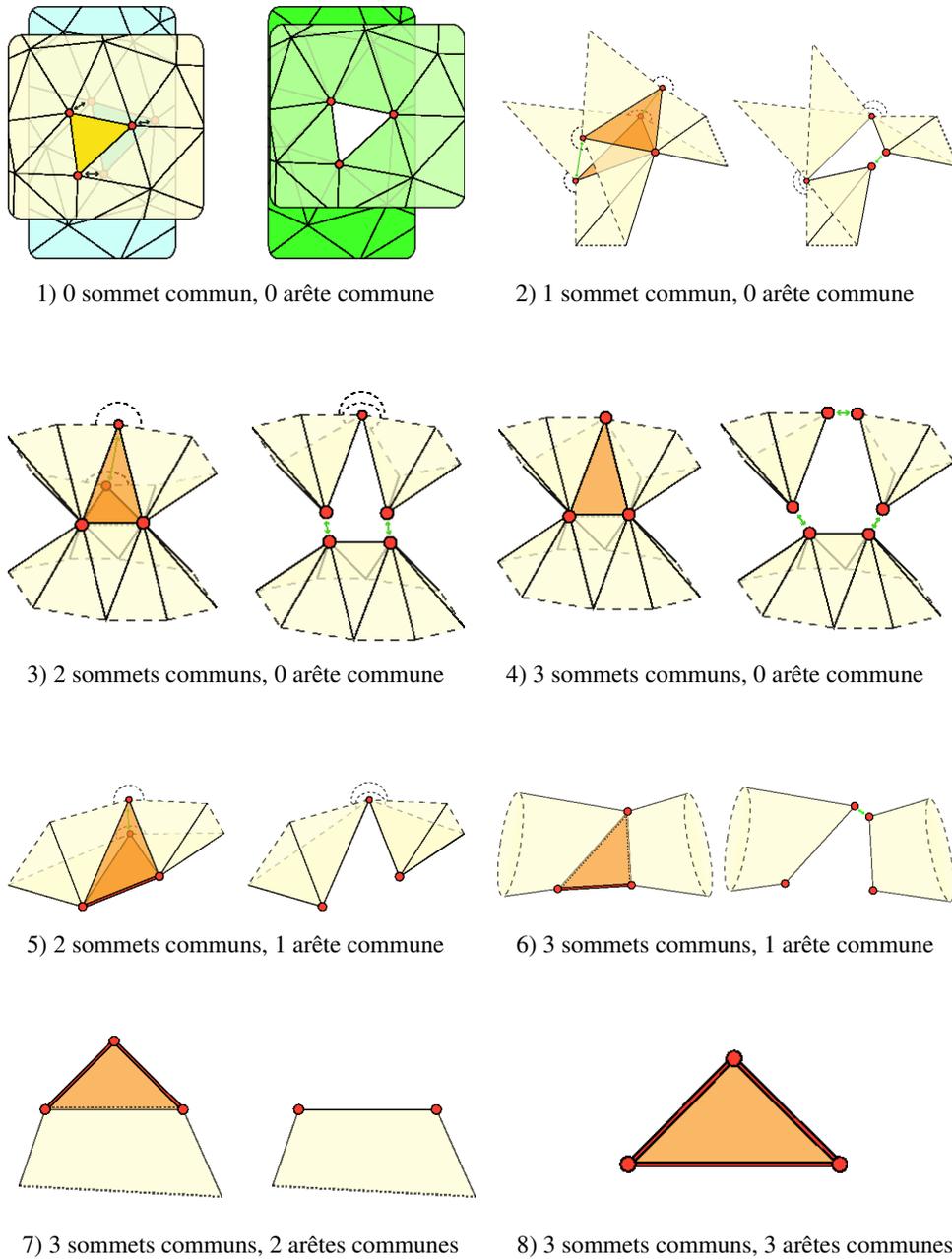


FIG. 2.5 – Les 8 configurations possibles d'effondrement de deux facettes couplées au cours du processus de convection géométrique (schémas tirés de [52]). Les différentes configurations sont caractérisées par le nombre de sommets et d'arêtes partagés par les deux facettes dont l'effondrement est considéré. Les configurations 1, 3, 4, 6 et 8 conduisent à des changements topologiques de type création de poignées ou apparition/disparition de composantes connexes.

Algorithme 1 Convection_geometrique(ensemble de points P)

```

initialiser la surface  $\hat{S}$  sur l'enveloppe convexe de  $P$  ;
pour chaque facette orientée  $pqr$  de  $\hat{S}$  faire
  si  $pqr$  ne vérifie pas le prédicat  $\mathbf{P}_{go}$  ou bloque une poche alors
    si les facettes orientées  $pqr$  et  $qpr$  appartiennent à  $\hat{S}$  alors
      supprimer  $pqr$  et  $qpr$  de  $\hat{S}$  ; {effondrement de deux facettes couplées}
    sinon
       $s \leftarrow \mathbf{Q}_{dt}(pqr)$  ; { $pqrs$  forme un tétraèdre de Delaunay}
      remplacer  $pqr$  par  $pqs$ ,  $qrs$  et  $rps$  ;
    finsi
  restaurer la connectivité entre les facettes orientées ;
finsi
fin pour
retourner  $\hat{S}$  ;

```

Garanties La terminaison de l'algorithme a été prouvée par Chaine [52]. Le résultat d'une reconstruction est également assuré, par construction, d'être une 2-variété combinatoire. En revanche, il n'a pas été prouvé que l'algorithme de convection géométrique produit une surface homéomorphe à la surface échantillonnée sous certaines conditions d'échantillonnage. Cependant, l'algorithme repose sur des propriétés similaires à celles du Flow Complex, pour lequel ce résultat a été démontré pour un ε -échantillon localement uniforme en entrée [80].

2.3 Propriétés utiles en vue d'une approche dynamique

Rappelons que notre objectif est de se doter d'un cadre flexible de reconstruction de surface, d'une part en simplifiant les données à la volée, et d'autre part en offrant la possibilité de mettre à jour la surface reconstruite de manière dynamique et incrémentale. L'algorithme de convection géométrique offre des propriétés intéressantes dans ce sens. Tout d'abord, la localité et l'indépendance des opérations qui permettent de faire converger la surface font que l'algorithme ne nécessite pas forcément de construire une triangulation de Delaunay globale. C'est un avantage important pour traiter des données dont on ne retiendra qu'une fraction. La manière dont se construisent les facettes de la surface définitive est en outre favorable à la simplification de l'échantillon de points dans la mesure où celle-ci correspond localement à l'avancée d'un front. La surface n'est ainsi pas découverte de manière désordonnée, mais par patches, ce qui constitue une propriété intéressante pour contrôler la distribution de l'échantillon dans le cadre de la simplification. Nous allons discuter en détail de ce point dans le paragraphe suivant. Concernant l'aspect dynamique et incrémental, le fait que l'algorithme ne repose pas sur une heuristique globale permet d'envisager un algorithme de mise à jour relançant le processus de convection géométrique localement suite à une modification des données. Dans le chapitre 5, nous mettrons en évidence qu'une forme d'historique du processus de reconstruction peut être stockée dans la triangulation de Delaunay des points retenus pour permettre la mise à jour locale.

Analyse de la construction du graphe de voisinage La surface en convection comporte à chaque instant un certain ensemble de sommets, d'arêtes et de facettes. Une facette qui ne vérifie pas la propriété de Gabriel orientée ou bloquant une poche a une existence temporaire ; sinon elle est dite terminale. Les arêtes de facettes terminales sont elles mêmes des arêtes terminales, dont la formation est préalable à celles de facettes terminales.

Nous expliquons ici en quoi la convection géométrique établit localement des relations de voisinage surfaciques, i.e. des arêtes terminales, selon un processus d'expansion de surface. Les nouveaux sommets peuvent être créés de deux manières différentes. La surface en convection commence par atteindre des points de l'échantillon dispersés sur la surface. Nous désignons par *germes* ce type de points isolés, qui apparaissent aussi à chaque fois que de nouvelles concavités sont rencontrées par la surface en convection. Ainsi, un nouveau germe est créé à chaque fois qu'une cellule de Delaunay correspondant à un embranchement de l'axe médian externe est ouverte. Lorsque la surface converge sans rencontrer de tel embranchement, de nouveaux points sont créés autour des germes. Chaque germe est à l'origine d'un front de propagation de relations d'adjacences entre des points voisins. Le graphe de voisinage est ainsi construit depuis plusieurs sommets germes en même temps. D'autres relations de voisinage terminales résultent de la fusion de deux fronts de propagation.

On a donc un processus qui, à l'échelle locale, alterne dissémination de points germes et conquête de la surface autour d'eux. Ce principe est illustré par un exemple en deux dimensions sur la figure 2.6. La figure 2.7 illustre ces deux aspects sur deux modèles reconstruits. Des mesures montrent qu'en moyenne près de 90% des facettes de la surface en convection qui ne vérifient pas la propriété de Gabriel orientée s'ouvrent vers un point formant une arête de la surface finale avec l'un de leurs sommets. Dans les autres cas, les facettes sont donc ouvertes vers des points germes.

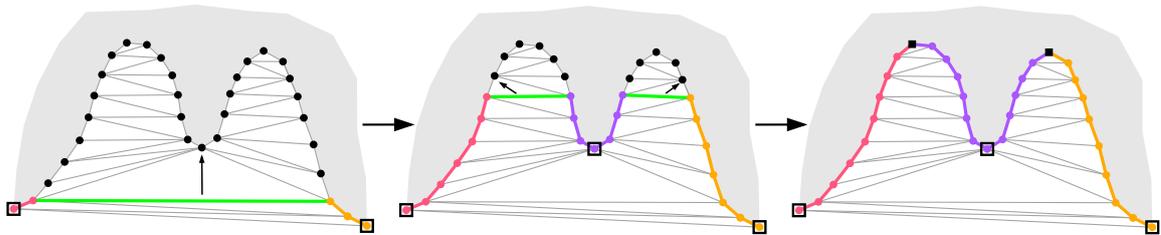
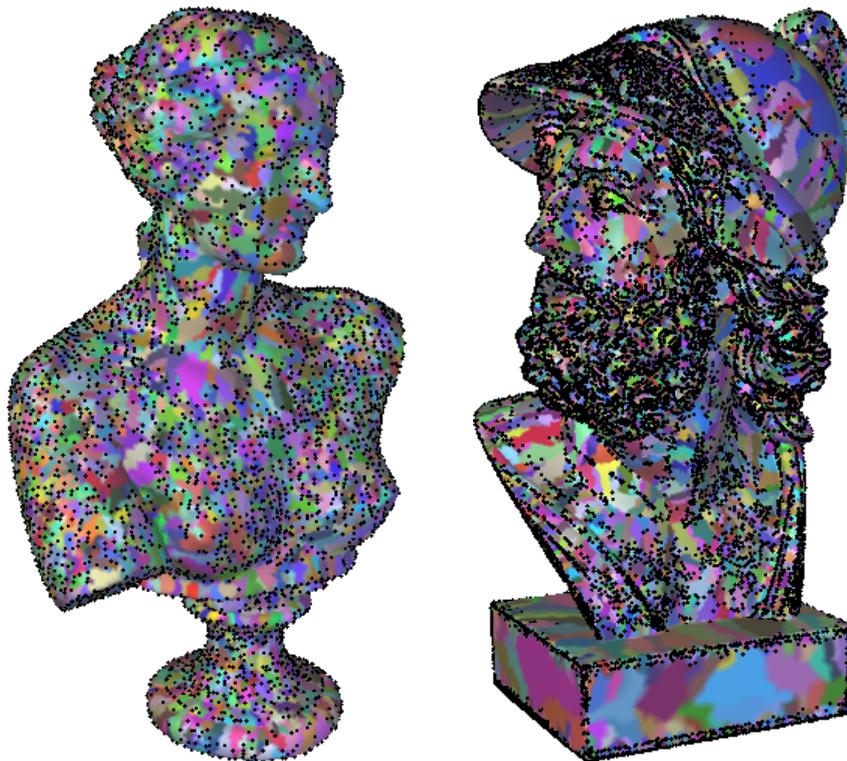


FIG. 2.6 – Construction du graphe de voisinage sur une courbe au cours de trois étapes d'un processus de reconstruction par convection géométrique en 2D. Les points germes sont entourés d'un carré, et chaque front de propagation créé apparaît dans une couleur différente. Les différents fronts fusionnent au fond des concavités (points noirs).



Modèle	$ P $	Nombre de points germes	Nombre de facettes ouvertes vers des points voisins
APHRODITE (à gauche)	46 096	7 426	86 %
AJAX (à droite)	273 397	30 626	88 %

FIG. 2.7 – Illustration de la manière dont s’est développée la surface pour plusieurs modèles reconstruits. Pour chaque modèle sont représentés les points germes et les régions de la surface associées à chacun d’eux, représentées par des couleurs aléatoires. Le tableau indique le nombre de germes pour chaque modèle et le pourcentage de facettes de la surface en convection qui se sont ouvertes vers des points formant une arête terminale avec l’un de leurs trois sommets.

Mesure d'épaisseur locale

Parallèlement à la reconstruction, notre second objectif est de simplifier l'échantillon de points en tenant compte de la géométrie locale tout en maintenant des conditions d'échantillonnage favorables pour l'algorithme de reconstruction. Cela nécessite une mesure permettant de caractériser localement la densité d'échantillonnage requise pour approximer la surface à reconstruire. Cette mesure doit refléter à la fois la courbure de la surface et l'épaisseur du solide qu'elle borne. Des travaux réalisés dans la communauté de géométrie algorithmique ont en effet mis en évidence que les conditions optimales d'échantillonnage d'une surface dans le cadre de la reconstruction de surface sont liées à ces deux aspects [13, 53]. En particulier, la notion d' ε -échantillon pour des surfaces lisses est définie à partir de la distance à l'axe médian, ou local feature size (LFS) [13]. Un inconvénient du LFS est que l'axe médian est très instable par rapport aux détails de la forme de la surface. Cela signifie que de petites variations dans la forme peuvent conduire à des axes médians très différents. Ainsi, dans le cadre discret, l'ensemble des pôles extraits du diagramme de Voronoï d'une surface échantillonnée est très instable par rapport au bruit. Plusieurs approches ont été proposées pour tenter de résoudre ce problème [21]. Chazal et Lieutier [53] ont introduit la notion de λ -axe médian, qui permet de filtrer l'axe médian pour ne le prendre en compte qu'au delà d'un certain niveau de détail. Cependant, la construction d'une triangulation de Delaunay globale est nécessaire pour rendre l'estimation du LFS possible.

Nous proposons dans ce chapitre une solution alternative de mesure d'épaisseur avec une définition locale et une estimation simple à partir d'un échantillon de points. Nous précisons les conditions d'échantillonnage dans lesquelles cette mesure peut être estimée dans le cadre discret.

3.1 Cadre continu

Soit S une surface fermée. Notre objectif est d'estimer en tout point $\mathbf{x} \in S$ l'épaisseur locale du solide borné par S . Le principe de notre mesure est de faire croître une boule \mathcal{B} autour de \mathbf{x} jusqu'à ce qu'un critère lié à la courbure soit vérifié par $\mathcal{B} \cap S$, ou bien que \mathcal{B} rencontre S , ce qui a pour conséquence que $\mathcal{B} \cap S$ n'est plus un disque topologique. Le rayon de cette boule constitue l'épaisseur locale au point \mathbf{x} . Ce principe donne lieu à la définition mathématique suivante.

Définition 1 (Épaisseur locale). *Étant donné une surface S fermée, l'épaisseur locale $e(\mathbf{x})$ en un point*

$\mathbf{x} \in S$ pour une précision géométrique ρ_{geom} fixée du solide borné par S est définie par :

$$e : S \rightarrow \mathbb{R}_+, \quad \mathbf{x} \mapsto e(\mathbf{x}) = \min\{r_{geom}(\mathbf{x}), r_{topo}(\mathbf{x})\}$$

où $r_{geom}(\mathbf{x})$ et $r_{topo}(\mathbf{x})$ sont respectivement appelés rayon géométrique et rayon topologique en \mathbf{x} . Soit $\mathbf{n}(\mathbf{x})$ un vecteur normal unitaire en un point \mathbf{x} (peu importe son orientation). Ces deux rayons sont définis comme suit :

- $r_{geom}(\mathbf{x})$ est la distance du point \mathbf{x} au point $\mathbf{y} \in S$ le plus proche tel que :

$$|\mathbf{n}(\mathbf{y}) \cdot \mathbf{n}(\mathbf{x})| \leq \rho_{geom} \quad \rho_{geom} \in [0,1]$$

- $r_{topo}(\mathbf{x})$ est la distance du point \mathbf{x} au point $\mathbf{y} \in S$ le plus proche tel que $\mathcal{B}(\mathbf{x}, \|\mathbf{xy}\|) \cap S$ n'est pas un disque topologique.

La proximité entre deux points sur la surface S est entendue ici au sens de la distance euclidienne. Le rayon r_{geom} reflète l'épaisseur dans les régions où la surface est fortement courbée. Les courbures ne sont pas estimées explicitement ; nous étudions les variations de la direction des normales pour constater qu'un voisinage sur la surface est plus ou moins fortement courbé, ce qui évitera des calculs complexes dans le cadre discret. Le rayon géométrique est ainsi calculé comme la distance au point le plus proche tel que la déviation de la direction de la normale dépasse un certain seuil. Il est à noter que la vérification de ce critère n'implique pas la connaissance d'une orientation consistante globale des vecteurs normaux, mais uniquement de leur direction, ce qui constitue une information locale. Le paramètre ρ_{geom} correspond à un seuil de *contraste* que l'on souhaite préserver dans les détails en estimant l'épaisseur, qui détermine un intervalle de variation tolérée de la direction de la normale par rapport à celle du point de mesure. Il est ainsi possible de tenir compte de toutes les aspérités de la surface, ou au contraire de considérer les variations de la géométrie à une échelle plus globale, ce qui peut être important en pratique pour traiter des données bruitées. On peut noter que sur une surface non lisse, r_{geom} s'annule au niveau d'une arête vive d'angle supérieur à $\arccos(\rho_{geom})$.

Le rayon r_{topo} limite la mesure à un disque topologique sur la surface. Il contrôle ainsi l'épaisseur du point de vue de la proximité entre la nappe de la surface sur laquelle se trouve le point de mesure et la nappe la plus proche. Sous l'hypothèse que S est une surface lisse, le rayon $r_{topo}(\mathbf{x})$ en un point \mathbf{x} peut être aussi défini comme la distance du point \mathbf{x} au point $\mathbf{y} \in S$ le plus proche tel que la boule $\mathcal{B}(\mathbf{x}, r_{topo}(\mathbf{x}))$ est tangente à S , i.e. $|\mathbf{n}(\mathbf{y}) \cdot \frac{\mathbf{xy}}{\|\mathbf{xy}\|}| = \rho_{topo}$ avec $\rho_{topo} = 1$. Cette définition permet de calculer r_{topo} de manière analytique, et de l'effectuer avec une certaine tolérance pour des valeurs de ρ_{topo} inférieures à 1. Ceci permet aussi d'élaborer un algorithme dans le cadre discret en considérant la direction des normales.

La figure 3.1 (a) illustre la mesure de l'épaisseur locale en deux points d'une courbe plane. Au point \mathbf{x}_1 , le rayon minimal est le rayon topologique, tandis qu'au point \mathbf{x}_2 , le rayon minimal est le

rayon géométrique. La figure 3.1 (b) met en évidence les différentes portions de la courbe sur lesquelles l'épaisseur locale est contrôlée par le rayon géométrique ou par le rayon topologique, ce qui permet de définir une segmentation en région fortement courbées et en régions minces.

Remarque sur le sens du paramètre ρ_{geom} Soulignons que la tolérance impliquée dans le calcul du rayon géométrique n'est pas liée à une échelle de forme, puisqu'elle est indépendante d'une métrique. Elle peut être davantage interprétée comme un niveau de précision auquel est mesurée l'épaisseur locale. Dans le cadre de la reconstruction sélective, c'est aussi la précision avec laquelle sera déterminée la densité d'échantillonnage requise dans les régions fortement courbées. Par la suite, nous parlerons simplement de niveau de précision pour caractériser le rôle de ce paramètre.

Remarque sur la dépendance des deux rayons Il est à noter que les deux rayons ne sont pas forcément pertinents indépendamment l'un de l'autre. Il se peut par exemple que le calcul du rayon géométrique soit influencé par la proximité de la nappe de la surface la plus proche de celle sur laquelle se trouve le point de mesure. Dans ce cas, le rayon topologique prend automatiquement le relais. Inversement, ce dernier peut mesurer une épaisseur qui ne reflète pas suffisamment la géométrie locale lorsque le critère est vérifié pour un point éloigné, auquel cas le rayon géométrique sera automatiquement le plus pertinent.

Remarque sur le domaine de validité de la définition Les rayons géométrique et topologique, et par conséquent la mesure d'épaisseur locale, sont définis pour toute surface fermée (ou courbe fermée en 2D). On peut cependant étendre la définition aux surfaces ouvertes sous certaines conditions. Dans le cas de surfaces ouvertes, r_{geom} ou r_{topo} peuvent localement être considérés comme infinis s'il n'existe pas de points y satisfaisant les définitions. Il est facile de vérifier que le rayon géométrique est fini en tout

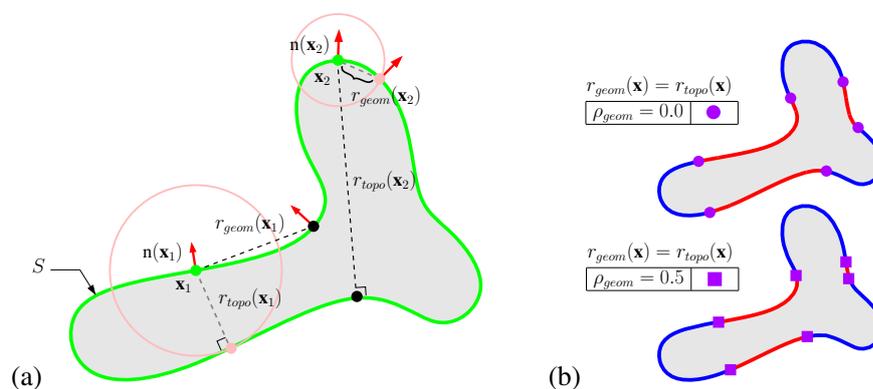


FIG. 3.1 – Épaisseur locale mesurée sur une courbe plane lisse. En (a), le rayon des cercles représente l'épaisseur e en deux points \mathbf{x}_1 et \mathbf{x}_2 . En (b), les points vérifient $r_{geom}(\mathbf{x}) = r_{topo}(\mathbf{x})$ pour deux valeurs fixées de ρ_{geom} . Sur les portions de courbe bleues, on a $e(\mathbf{x}) = r_{geom}(\mathbf{x})$ pour tout point \mathbf{x} de la courbe, et sur les portions de courbe rouge, on a $e(\mathbf{x}) = r_{topo}(\mathbf{x})$.

point d'une surface S si et seulement si

$$\rho_{geom} \geq \min_{\mathbf{x}, \mathbf{y} \in S} (|\mathbf{n}(\mathbf{x}) \cdot \mathbf{n}(\mathbf{y})|)$$

Dans le cas contraire, r_{geom} est infini en tout point de S . On peut noter que si S est fermée, alors $\min_{\mathbf{x}, \mathbf{y} \in S} (|\mathbf{n}(\mathbf{x}) \cdot \mathbf{n}(\mathbf{y})|) = 0$. Dans le cas de surfaces ouvertes, le rayon topologique peut également être infini en certains points si le critère le définissant n'est pas vérifié.

Exemple La figure 3.2 illustre la mesure d'épaisseur en deux dimensions dans le cas particulier d'une parabole. Nous avons tracé le graphe des fonctions r_{geom} , r_{topo} et e définies en tout point d'une parabole d'équation $y = x^2$ pour différentes valeurs de ρ_{geom} . Sur cette courbe ouverte, le rayon topologique n'est pas défini pour $x \in [-\sqrt{2}, \sqrt{2}]$; il est considéré comme infini dans cet intervalle. On peut constater que le rayon géométrique suit globalement les variations de la parabole pour les différentes valeurs de ρ_{geom} , avec apparition d'un pic au point de courbure maximum en $(0, 0)$, dont l'amplitude augmente à mesure que la valeur de ρ_{geom} diminue. Par conséquent, pour que r_{geom} reflète au mieux la courbure locale, la valeur du paramètre ρ_{geom} doit rester suffisamment proche de 1.

La figure 3.3 met en parallèle le local feature size (en pointillés) et la demi-épaisseur locale calculée comme $\frac{1}{2}e$, mettant en évidence une certaine proximité des deux mesures. Nous n'avons cependant pas de résultat général permettant de mettre en relation l'épaisseur locale e et le local feature size. La nature globale du LFS, par rapport à l'approche plus locale que nous adoptons, rend difficile l'établissement d'une telle relation et l'utilisation des résultats d'Amenta et Bern [13].

Remarque sur le rayon lipschitzien Des idées similaires mais plus formelles ont également été proposées par Boissonnat et Oudot [45], qui ont proposé une nouvelle théorie de l'échantillonnage de surface reposant sur une forme de mesure d'épaisseur locale. Un lien direct avec notre travail est la définition en tout point d'un rayon tenant compte des variations locales de la surface. Plus précisément, étant donné une surface S , les auteurs définissent le rayon k -lipschitzien en un point \mathbf{x} comme le rayon de la plus grande boule \mathcal{B} centrée en \mathbf{x} telle que $\mathcal{B} \cap S$ est le graphe d'une fonction k -lipschitzienne de deux variables. Ils prouvent que des garanties similaires à celles obtenues avec des ε -échantillons pour des surfaces lisses peuvent être obtenues à partir d'un échantillon d'une surface k -lipschitzienne S telle que tout point de S possède un point de l'échantillon à une distance inférieure à une fraction du rayon lipschitzien de S . Les auteurs n'ont pas précisé les moyens à mettre en œuvre pour estimer les rayons, qui seront discutés dans une version étendue de leur article. Pour notre part, nous nous sommes davantage attachés à l'estimation de l'épaisseur dans le cadre discret, aussi nous avons cherché à proposer une définition simple pouvant donner lieu à un algorithme applicable à un échantillon de points.

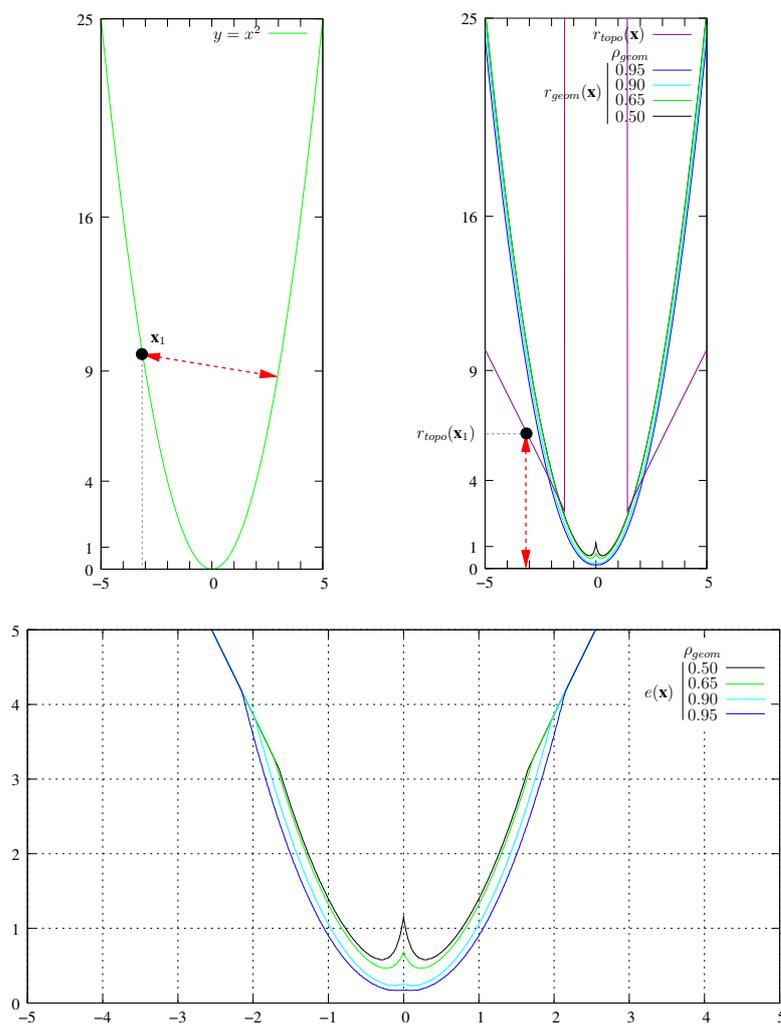


FIG. 3.2 – Variation du rayon géométrique r_{geom} , du rayon topologique r_{topo} et de l'épaisseur locale e sur une parabole d'équation $y = x^2$.

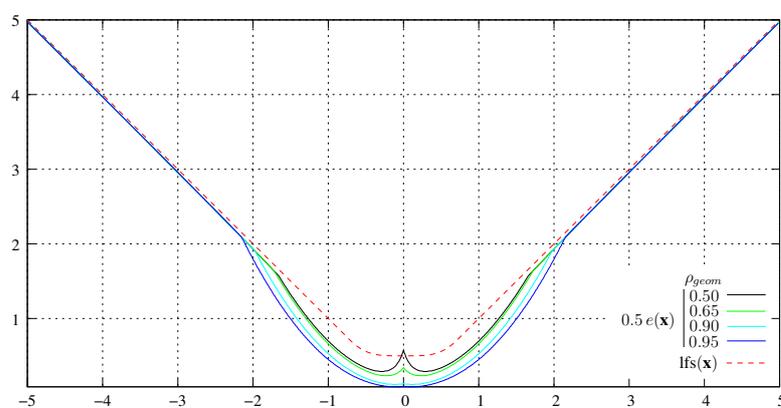


FIG. 3.3 – Lien entre la demi-épaisseur locale et le LFS observé sur la parabole d'équation $y = x^2$.

3.2 Cadre discret

La mesure d'épaisseur locale peut être étendue directement au cadre d'un échantillon de points fini P sur une surface S supposée lisse, dont on connaît la direction des vecteurs normaux.

Définition 2 (Épaisseur locale discrète). *Étant donné un échantillon P de points sur une surface fermée lisse S , l'épaisseur locale discrète \tilde{e} en un point $\mathbf{p} \in P$ pour des précisions ρ_{geom} et ρ_{topo} fixées est définie par :*

$$\tilde{e} : P \rightarrow \mathbb{R}_+, \quad \mathbf{p} \mapsto \tilde{e}(\mathbf{p}) = \min\{\tilde{r}_{geom}(\mathbf{p}), \tilde{r}_{topo}(\mathbf{p})\}$$

où les rayons $\tilde{r}_{geom}(\mathbf{p})$ et $\tilde{r}_{topo}(\mathbf{p})$ au point \mathbf{p} sont définis comme suit :

- $\tilde{r}_{geom}(\mathbf{p})$ est la distance du point \mathbf{p} au point $\mathbf{q} \in P$ le plus proche tel que :

$$|\mathbf{n}(\mathbf{q}) \cdot \mathbf{n}(\mathbf{p})| \leq \rho_{geom} \quad \rho_{geom} \in [0,1]$$

- $\tilde{r}_{topo}(\mathbf{p})$ est la distance du point \mathbf{p} au point $\mathbf{q} \in P$ le plus proche tel que :

$$|\mathbf{n}(\mathbf{q}) \cdot \frac{\mathbf{pq}}{\|\mathbf{pq}\|}| \geq \rho_{topo} \quad \rho_{topo} \in [0,1]$$

La définition est similaire à celle de la mesure dans le cadre continu, le rayon topologique discret \tilde{r}_{topo} étant estimé avec une certaine tolérance réglée par le paramètre ρ_{topo} , de même que pour la définition paramétrée de ce rayon dans le cadre continu. La validité de cette définition et sa relation avec le cadre continu dépendent de certaines propriétés de P par rapport à S . Dans la section suivante, nous proposons d'exhiber des conditions d'échantillonnage pour lesquelles l'épaisseur locale discrète est bien définie.

3.3 Conditions d'échantillonnage

Un échantillon de points P sur une surface lisse fermée S doit vérifier certaines propriétés pour que la définition de l'épaisseur locale dans le cadre discret ait un sens par rapport à la même mesure dans le cadre continu. Ces propriétés doivent permettre de limiter le rayon calculé dans le cadre discret en fonction des caractéristiques de la surface échantillonnée. Nous avons en particulier déterminé des conditions d'échantillonnage pour lesquelles la mesure d'épaisseur discrète est bornée par rapport à celle calculée dans le cadre continu à partir de la surface échantillonnée. Nous définissons ces conditions d'échantillonnage indépendamment pour le rayon géométrique et pour le rayon topologique. La figure 3.4 fournit une illustration pour chacune. Dans ce paragraphe, nous noterons r_{geom}^a (resp. \tilde{r}_{geom}^a) le rayon géométrique continu (resp. discret) paramétré par une valeur $a \in [0,1]$, et r_{topo}^b (resp. \tilde{r}_{topo}^b) le rayon topologique continu (resp. discret) paramétré par une valeur $b \in [0,1]$.

Définition 3 (ρ_{geom} -échantillon). *Soit P un échantillon de points sur une surface lisse fermée S . P est un ρ_{geom} -échantillon de S paramétré par $a \in [0,1]$ s'il existe une constante $C \geq 1$ telle que pour tout point $\mathbf{x} \in S$, s'il existe un point $\mathbf{y} \in S$ tel que $\arccos(|\mathbf{n}(\mathbf{x}) \cdot \mathbf{n}(\mathbf{y})|) > C \arccos(a)$, alors il existe un point $\mathbf{p} \in P$ tel que $\|\mathbf{xp}\| \leq \|\mathbf{xy}\|$ et $|\mathbf{n}(\mathbf{x}) \cdot \mathbf{n}(\mathbf{p})| \leq a$.*

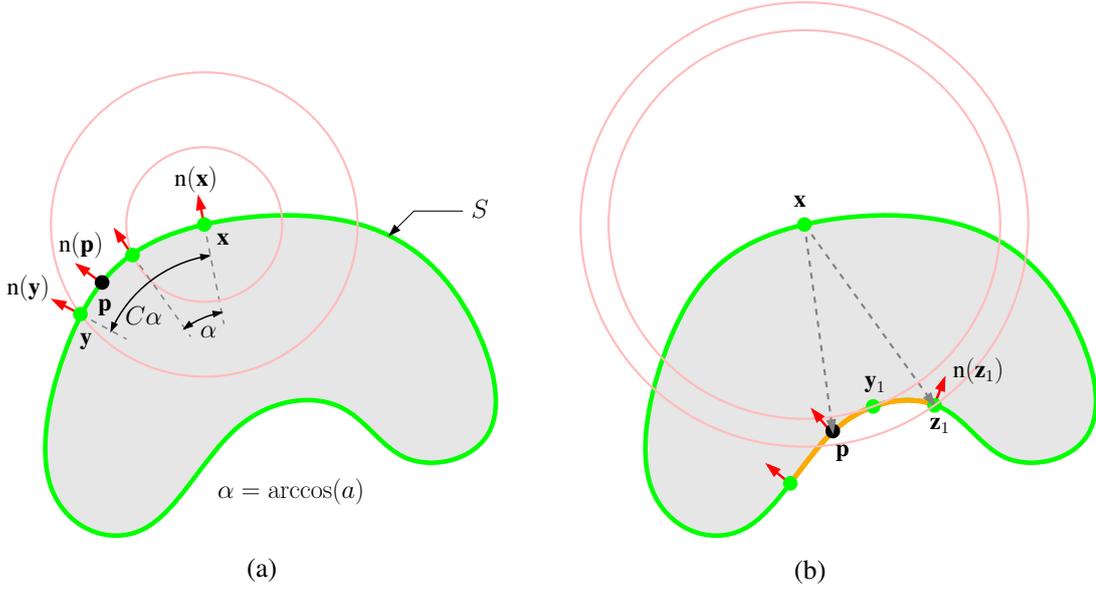


FIG. 3.4 – En (a) : illustration de la condition de ρ_{geom} -échantillon paramétré par $a \in [0, 1]$; en (b) : illustration de la condition de ρ_{topo} -échantillon.

Lemme 1. Si P est un ρ_{geom} -échantillon d'une surface lisse fermée paramétré par $a \in [0, 1]$, alors il existe une constante $C \geq 1$ telle que pour tout point $\mathbf{p} \in P$ on ait :

$$\tilde{r}_{geom}^a(\mathbf{p}) < r_{geom}^{\cos(C \arccos(a))}(\mathbf{p})$$

La preuve du lemme 1 est directe par reformulation de la définition 3. On notera que $a > \cos(C \arccos(a))$. Si P est un ρ_{geom} -échantillon paramétré par une valeur $a \in [0, 1]$, alors le rayon géométrique discret mesuré sur cet échantillon minore celui que l'on obtiendrait dans le cadre continu avec une valeur $a' = \cos(C \arccos(a))$ légèrement moins stricte du paramètre spécifiant la déviation maximale de la direction des normales (figure 3.4(a)).

Définition 4 (ρ_{topo} -échantillon). Soit P un échantillon de points sur une surface lisse fermée S . P est un ρ_{topo} -échantillon de S paramétré par $b \in [0, 1]$ si pour tout point $\mathbf{x} \in S$, s'il existe un point \mathbf{y} tel que $\mathcal{B}(\mathbf{x}, \|\mathbf{xy}\|) \cap S$ n'est pas un disque topologique, alors il existe un point $\mathbf{p} \in P$ tel que $\mathcal{B}(\mathbf{x}, \|\mathbf{xp}\|) \cap S$ n'est pas un disque topologique et $\|\mathbf{xp}\| \leq \|\mathbf{xz}\|$, où $\mathbf{z} \in S$ le point le plus proche de \mathbf{y} tel que $|\mathbf{n}(\mathbf{z}) \cdot \frac{\mathbf{xz}}{\|\mathbf{xz}\|}| < b$.

Lemme 2. Si P est un ρ_{topo} -échantillon d'une surface lisse fermée paramétré par $b \in [0, 1]$, alors pour tout point $\mathbf{p} \in P$, on a :

$$\tilde{r}_{topo}^b(\mathbf{p}) < r_{topo}^b(\mathbf{p})$$

La démonstration du lemme 2 est immédiate en reformulant la définition 4. Sur une surface lisse fermée S , sans perte de généralité, considérons en un point $\mathbf{x} \in S$ fixé le premier point $\mathbf{y}_1 \in S$ sur une

nappe de la surface autre que celle sur laquelle se trouve \mathbf{x} en faisant croître une boule \mathcal{B} autour de \mathbf{x} , i.e. le point où se réalise le rayon topologique continu. Si P est un ρ_{topo} -échantillon de S , cela signifie qu'un point de P vient s'interposer entre \mathbf{y}_1 et le point \mathbf{z}_1 le plus proche de \mathbf{y}_1 pour lequel le critère de détection discrète de tangence entre \mathcal{B} et S n'est plus vérifié (figure 3.4(b)).

Ces conditions d'échantillonnage impliquent une densité minimale de l'échantillon d'après les paramètres spécifiant les ρ_{geom} - et ρ_{topo} -échantillons, mais elles n'imposent pas de borne supérieure, i.e. l'échantillon peut être aussi dense que souhaité dès lors que la densité minimale est atteinte, sans affecter l'estimation de l'épaisseur locale. En particulier, la distance relative des points entre eux et leur distribution ne sont pas contraintes, ce qui permet de traiter des échantillons de points non uniformes.

3.4 Évaluation

Le rayon topologique et le rayon géométrique peuvent être évalués de manière analytique sur une courbe ou une surface dont on connaît une représentation explicite ou paramétrique. Nous nous intéressons plus spécifiquement à l'évaluation de la mesure discrète à partir d'un échantillon de points.

L'évaluation de \tilde{e} nécessite de connaître la direction de la normale à la surface en chaque point de l'échantillon, ou une estimation si cette information n'est pas fournie. Pour être cohérent avec le principe de localité de la mesure d'épaisseur, cette estimation peut être réalisée de manière locale à partir des k plus proches voisins (ce point fait l'objet d'une discussion dans l'annexe B).

D'un point de vue algorithmique, le moyen le plus simple de calculer $\tilde{e}(\mathbf{p})$ en un point $\mathbf{p} \in P$ est de faire appel à une requête géométrique \mathbf{Q}_{pv} qui retourne ses plus proches voisins de manière incrémentale. L'algorithme d'évaluation consiste alors à découvrir les voisins de \mathbf{p} par ordre croissant de leur distance à \mathbf{p} , jusqu'à ce qu'un des critères correspondant au calcul des rayons $\tilde{r}_{geom}(\mathbf{p})$ et $\tilde{r}_{topo}(\mathbf{p})$ soit vérifié. La distance au premier point qui permet de calculer l'un de ces rayons est l'épaisseur $\tilde{e}(\mathbf{p})$.

En pratique, une question importante concerne le choix des valeurs des paramètres ρ_{geom} et ρ_{topo} . Dans le cadre de notre méthode de reconstruction sélective, la valeur de ρ_{geom} est fixée par l'utilisateur. Ce paramètre permettra de contrôler le niveau de redondance toléré. La valeur de ρ_{topo} devrait être fixée d'après les conditions d'échantillonnage. Cependant, étant donné un échantillon de points, il est difficile de vérifier si celui-ci satisfait ces conditions. Dans l'ensemble de nos tests réalisés sur des échantillons très denses, nous avons utilisé $\rho_{topo} = 0.9$. Des études complémentaires sont nécessaires pour déterminer une méthode pour calculer ρ_{topo} pour un échantillon donné, de manière globale ou locale, ainsi que les relations entre les rayons géométriques et topologiques.

Il va de soit que pour chaque mesure, le nombre d'appels à la requête \mathbf{Q}_{pv} dépend de la géométrie de la surface et de la densité de l'échantillon de points. Pour une implantation efficace, une structure permettant de localiser rapidement les plus points les plus proches d'un point donné est nécessaire. Il peut s'agir par exemple d'un k d-arbre ou d'une triangulation de Delaunay. Nous avons implanté l'algorithme à l'aide d'un k d-arbre, moins coûteux à construire et à maintenir en mémoire qu'une triangulation de Delaunay globale pour de gros échantillons de points.

La figure 3.5 illustre les variations de l'épaisseur locale discrète sur un modèle représenté par un échantillon de points. La table 3.1 fournit des statistiques sur les performances de cette implantation pour divers modèles 3D représentés par des échantillons de points. Il est à noter que les temps de calcul pour des points sélectionnés de manière aléatoire ne sont pas nécessairement représentatifs des temps de reconstruction sélective, pour laquelle le choix des points d'évaluation est guidé par la géométrie locale.

Modèle		$\rho_{geom} = 0.65$			$\rho_{geom} = 0.9$		
Nom	$ P $	Temps	$\mathbf{Q}_{pv} (\times 10^6)$	\tilde{r}_{geom}	Temps	$\mathbf{Q}_{pv} (\times 10^6)$	\tilde{r}_{geom}
APHRODITE	46 096	26.9	24.2	93 %	10.6	7.9	98 %
CHAIR	59 124	58.0	54.9	94 %	26.5	22.8	98 %
TRIPLE HÉCATE	90 180	21.0	17.8	97 %	8.2	5.4	99 %
HAND	141 402	117.7	118.9	96 %	40.8	38.1	99 %
AJAX	273 397	18.5	13.6	95 %	8.1	4.6	99 %
DAVID	3 617 009	562.2	567.9	88 %	162.7	162.9	97 %

TAB. 3.1 – Statistiques sur l'évaluation de l'épaisseur locale pour divers modèles 3D. Pour chaque modèle, 10^6 évaluation ont été effectuées en des points choisis de l'échantillon choisis de manière aléatoire. Les temps de calcul sont donnés en secondes (calculs effectués sur un processeur Pentium IV 3 GHz avec 1 Go de RAM) ; les colonnes « $\mathbf{Q}_{pv} (\times 10^6)$ » correspondent au nombre de requêtes de plus proche voisin \mathbf{Q}_{pv} effectuées ; les colonnes « \tilde{r}_{geom} » correspondent au pourcentage du nombre d'évaluations où $\tilde{e}(\mathbf{x}) = \tilde{r}_{geom}(\mathbf{x})$.

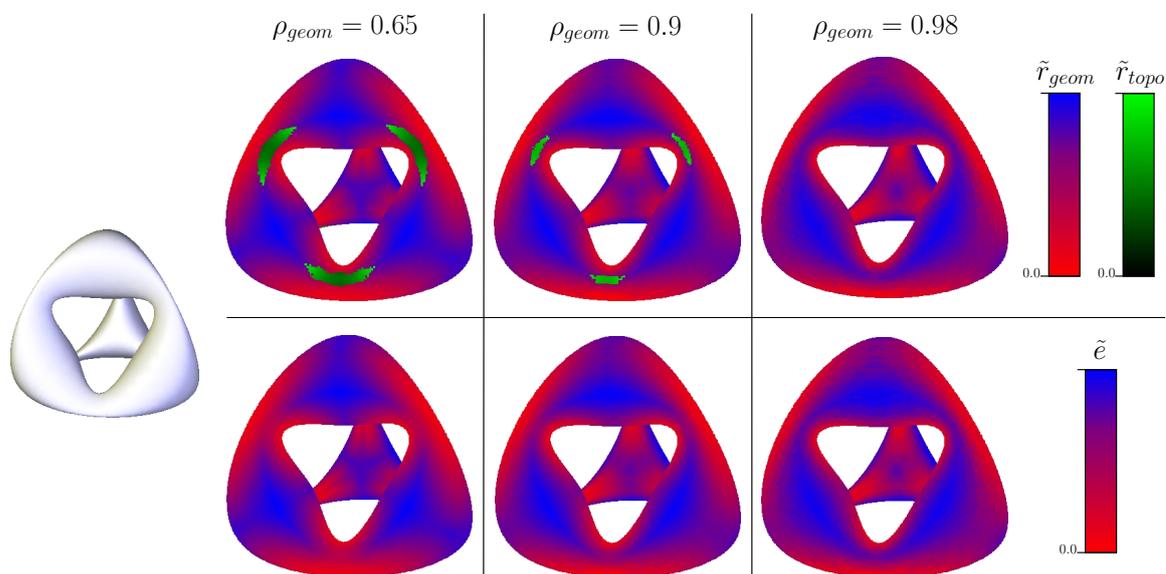


FIG. 3.5 – Visualisation des variations de l'épaisseur locale discrète pour différentes valeurs du paramètre ρ_{geom} sur le modèle CHAIR (59k points). La rangée supérieure distingue les régions de la surface où $\tilde{e}(\mathbf{p}) = \tilde{r}_{geom}(\mathbf{p})$ et $\tilde{e}(\mathbf{p}) = \tilde{r}_{topo}(\mathbf{p})$ par deux cartes de couleurs différentes, tandis que la rangée inférieure permet de visualiser la valeur de $\tilde{e}(\mathbf{p})$ en tout point.

Reconstruction sélective

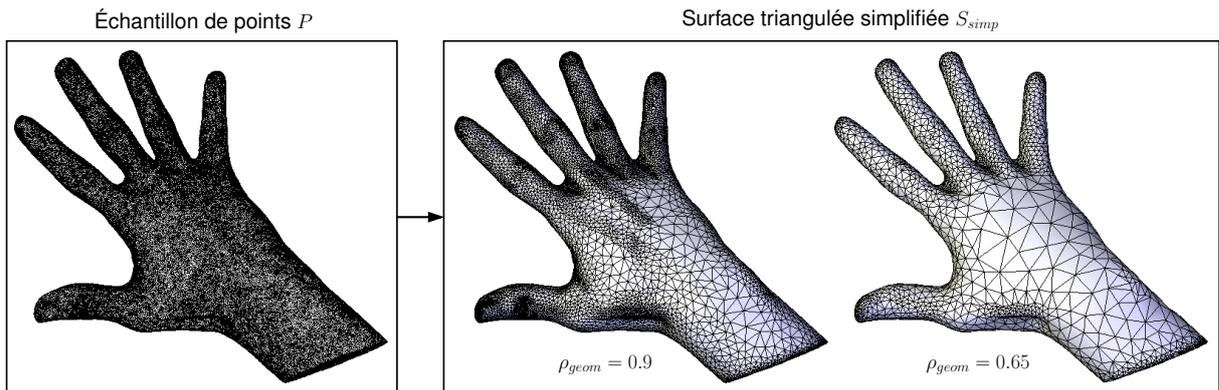


FIG. 4.1 – Reconstruction de surface sélective.

Dans ce chapitre, nous présentons un algorithme sélectif de reconstruction de surface qui permet de simplifier un échantillon de points non structuré très dense tout en reconstruisant une surface triangulée proche de la surface originale avec un contrôle du niveau de précision de la reconstruction (fig. 4.1). Notre algorithme prend en entrée un ensemble de points non structuré P échantillonné à partir d'une surface $S \in \mathbb{R}^3$, avec en chaque point la direction de la normale à la surface, sans besoin d'orientation consistante, et une tolérance de redondance caractérisée par un paramètre ρ_{geom} . La surface S est supposée lisse, et P suffisamment dense au sens où P forme un ε -échantillon pour une constante $\varepsilon > 0$ [13]. L'algorithme reconstruit une surface triangulée S_{simp} à partir de P dont les sommets s'appuient un sous-ensemble P_{simp} de P tel que $|P_{simp}| < |P|$.

Notre algorithme de reconstruction sélective associe l'algorithme de convection géométrique avec une procédure de sous-échantillonnage local éliminant à la volée les points jugés non significatifs relativement à la mesure d'épaisseur locale développée dans le chapitre précédent. Rappelons que cette mesure nécessite de connaître la direction de la normale à la surface en chaque point de P , mais pas son orientation. Si les normales ne sont pas fournies avec les données d'entrée, leur direction peut être estimée localement en approximant la surface par un plan tangent (annexe B).

Dans l'optique de reconstruire une surface en ne retenant qu'un sous-ensemble de points relativement à certains critères, nous ne construisons pas une triangulation de Delaunay globale de tous les points de l'échantillon en entrée. Nous avons développé un algorithme et une structure ad hoc combinant une structure de localisation spatiale de type k d-arbre avec une triangulation de Delaunay partielle des données qui permet réaliser le processus de reconstruction sélective de manière efficace tout en étant

moins complexe qu'une triangulation globale.

Nous commençons par décrire l'algorithme de reconstruction sélective dans la section 4.1 avant de présenter la manière dont est contrôlée la densité et la distribution de l'échantillon dans la section 4.2. Nous expliquons ensuite comment implanter l'algorithme de manière efficace dans la section 4.3. Dans la section 4.4, nous présentons enfin des résultats expérimentaux.

4.1 Algorithme de reconstruction sélective

L'algorithme de reconstruction sélective, dans son principe, met en œuvre le même schéma d'évolution de surface que l'algorithme de convection géométrique. La différence réside dans le fait qu'une partie des points de l'échantillon de départ, non encore impliquée dans surface reconstruite, est éliminée au cours du processus de reconstruction. Nous reprendrons par conséquent les notations du chapitre précédent en désignant par \hat{S} la surface au cours du processus de convection, par \mathbf{P}_{go} le prédicat correspondant à la propriété de Gabriel orientée et par \mathbf{Q}_{dt} la requête qui retourne un point de P formant un tétraèdre de Delaunay avec une facette orientée.

4.1.1 Procédure de simplification locale

À chaque fois qu'un nouveau point $\mathbf{p} \in P$ est incorporé dans la surface en convection \hat{S} , notre objectif est de supprimer les points de P dans un voisinage circulaire autour de \mathbf{p} dont le rayon reflète la géométrie locale. Pour cela, l'information géométrique portée par \mathbf{p} doit être suffisamment représentative de la position des points dans ce voisinage. Cette capacité de représentation est caractérisée par la mesure d'épaisseur locale introduite dans le chapitre précédent, utilisée comme une mesure de redondance locale. Ainsi, pour un point $\mathbf{p} \in P$ retenu, la procédure de simplification consiste à éliminer les points les plus proches dans P qui n'appartiennent pas à \hat{S} tant que leur distance à \mathbf{p} n'excède pas un rayon de simplification $r_{simp}(\mathbf{p})$ calculé à partir de $\tilde{e}(\mathbf{p})$. Nous définirons ce rayon dans la section 4.2, après avoir présenté l'algorithme de reconstruction.

4.1.2 Algorithme de convection géométrique sélective

L'ensemble des points retenus, i.e. incorporés dans la surface \hat{S} à un instant donné et conservés jusqu'à la fin du processus de reconstruction, sera noté $\hat{P} \subset P$. L'ensemble des points éliminés, complémentaire de \hat{P} par rapport à P , sera noté \hat{R} . Le prédicat \mathbf{P}_{go} et la requête \mathbf{Q}_{dt} seront désormais évalués dans $P \setminus \hat{R}$ à un instant donné. La requête \mathbf{Q}_{pv} sera toujours effectuée dans l'ensemble P pour obtenir la meilleure estimation possible de l'épaisseur locale. Les points supprimés doivent donc être perçus comme disqualifiés parmi les candidats aux futures insertions, plutôt que physiquement éliminés.

Nous décrivons maintenant l'algorithme de reconstruction sélective de manière complète. Celui-ci est schématisé en 2D dans la figure 4.2. La surface \hat{S} est d'abord initialisée en calculant une boîte

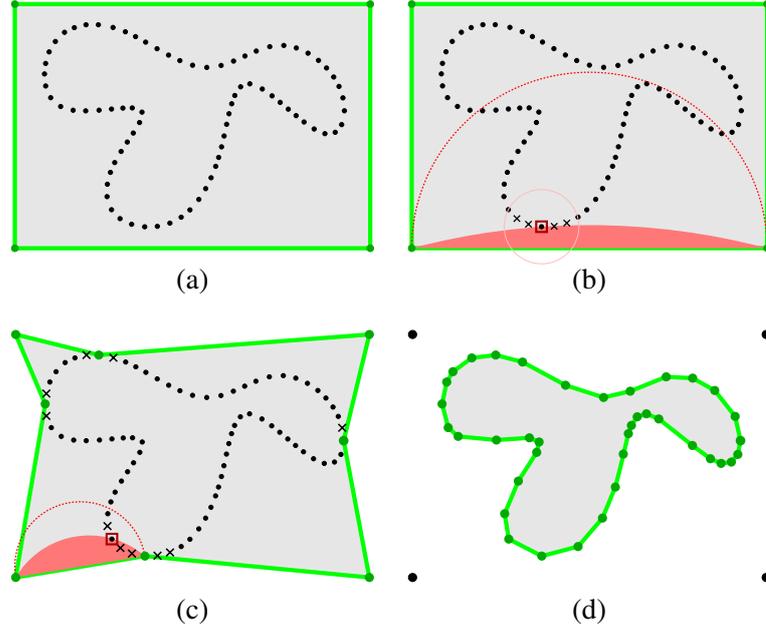


FIG. 4.2 – Reconstruction sélective sur un échantillon de points 2D. En (a), une courbe englobant l'échantillon de points est initialisée sur une boîte englobante. En (b), l'arête traitée a une demi-sphère de Gabriel non vide. Le point qui forme un triangle de Delaunay avec cette arête est recherché dans la demi-sphère de Gabriel associée (point entouré d'un carré). L'ensemble de points est alors simplifié localement autour du point retenu (les points en forme de croix sont éliminés). En (c), le point retenu a été inséré dans la triangulation et la courbe poursuit son évolution, jusqu'à obtenir le résultat final en (d).

englobante de P alignée selon les axes de coordonnées. La convection géométrique sélective est ensuite lancée à partir de cette surface, dont nous noterons \mathbf{pqr} une facette orientée vers l'intérieur de la surface. Nous notons \mathcal{B} la demi-sphère de Gabriel associée à la facette \mathbf{pqr} , et \mathcal{H} le demi-espace au-dessus du plan de la facette. La première étape consiste à vérifier si la facette \mathbf{pqr} satisfait le prédicat $\mathbf{P}_{\mathbf{go}}$.

- Si \mathcal{B} contient un point de $P \setminus \hat{R}$, alors le prédicat $\mathbf{P}_{\mathbf{go}}$ n'est pas satisfait par la facette \mathbf{pqr} . On vérifie alors si les deux facettes couplées \mathbf{pqr} et \mathbf{qpr} appartiennent à \hat{S} . Si c'est le cas, les facettes couplées s'effondrent et la connectivité est restaurée entre leurs facettes voisines. Sinon, on cherche le point $\mathbf{s} \in (P \setminus \hat{R}) \cap \mathcal{B}$ tel que \mathbf{pqrs} forme un tétraèdre de Delaunay dans $P \setminus \hat{R}$ (requête $\mathbf{Q}_{\mathbf{dt}}$). Si $\mathbf{s} \notin \hat{S}$, alors la procédure de simplification est appelée sur \mathbf{s} . Ce point est inséré dans \hat{S} et la facette \mathbf{pqr} est remplacée par les facettes \mathbf{pqs} , \mathbf{qrs} et \mathbf{rps} .
- Si la facette \mathbf{pqr} satisfait le prédicat $\mathbf{P}_{\mathbf{go}}$, alors on vérifie si celle-ci ne bloque pas une poche, ou bien est attachée à un sommet de la boîte englobante. Si la facette est considérée comme bloquant une poche ou attachée à un sommet de la boîte englobante, alors on cherche le point $\mathbf{s} \in (P \setminus \hat{R}) \cap \mathcal{H}$ tel que \mathbf{pqrs} forme un tétraèdre de Delaunay dans $P \setminus \hat{R}$ (requête $\mathbf{Q}_{\mathbf{dt}}$). Si $\mathbf{s} \notin \hat{S}$, la procédure de simplification est appelée sur \mathbf{s} et le point est inséré dans \hat{S} comme précédemment.

4.1.3 Détection des poches

Nous ne pouvons plus nous reposer sur une étude de la granularité locale pour détecter les poches, puisque celle-ci n'est fixée qu'à l'issue du processus de reconstruction. Nous avons donc instauré un nouveau critère, plus robuste, qui étudie d'une part la direction des normales aux sommets de la facette par rapport à celle de la normale à la facette, et d'autre part l'angle externe entre les sphères circonscrites aux deux tétraèdres de Delaunay incidents à la facette.

Le second critère est le principal. Celui-ci est inspiré du principe de classification des pôles internes et externes à la surface de l'algorithme Power Crust [15]. Si une facette bloque une poche, celle-ci intersecte en effet l'axe médian externe de la surface, et les sphères circonscrites aux deux tétraèdres de Delaunay incidents à cette facette sont deux sphères maximales voisines dont l'angle externe doit être grand (proche de π) si l'échantillon de points est suffisamment dense. Nous calculons comme suit le cosinus de l'angle externe γ entre la sphère circonscrite à \mathbf{pqrs} et la sphère circonscrite à la cellule de Delaunay adjacente par rapport à \mathbf{pqr} :

$$\cos(\gamma) = \frac{\|\mathbf{c}_1\mathbf{c}_2\|^2 - r_1^2 - r_2^2}{2r_1r_2}$$

en notant \mathbf{c}_1 et \mathbf{c}_2 les centres respectifs des deux sphères, et r_1 et r_2 leurs rayons. Si l'angle γ est supérieur à $\frac{\pi}{2}$, alors la facette \mathbf{pqr} est considérée comme bloquant une poche.

Ce critère nécessite d'effectuer la requête \mathbf{Q}_{dt} pour toutes les facettes qui vérifient le prédicat \mathbf{P}_{go} , en particulier pour les facettes terminales. Pour réduire le nombre de ces requêtes, nous effectuons un premier filtrage à l'aide du critère sur la direction des normales. Lorsqu'une facette bloque une poche, on peut observer que les vecteurs normaux en ses sommets sont presque orthogonaux au vecteur normal de la facette, sous l'hypothèse d'un échantillon suffisamment dense. Notons \mathbf{n}_f la normale unitaire d'une facette de la surface \hat{S} qui vérifie \mathbf{P}_{go} . Si la plus petite valeur entre $|\mathbf{n}_f \cdot \mathbf{n}(\mathbf{p})|$, $|\mathbf{n}_f \cdot \mathbf{n}(\mathbf{q})|$ et $|\mathbf{n}_f \cdot \mathbf{n}(\mathbf{r})|$ est inférieure à 0.5, alors seulement nous effectuons la requête \mathbf{Q}_{dt} sur la facette pour appliquer le critère sur l'angle externe entre les sphères circonscrites aux tétraèdres de Delaunay incidents à la facette. La figure 4.3 illustre le critère sur un exemple en deux dimensions.

Remarque sur l'initialisation de la surface Une solution alternative pour initialiser la surface consiste à construire une enveloppe convexe simplifiée en combinant l'algorithme Quickhull [25] avec la procédure de simplification. Cet algorithme applique en effet un schéma d'évolution de surface qui le rend propice à l'application de cette stratégie. La complexité de cet algorithme est $O(|P| \log |P|)$ en moyenne, et $O(|P|^2)$ dans le pire cas (cas d'un échantillon de points sur une surface convexe). La construction de la boîte englobante est plus simple et s'effectue en temps linéaire par rapport à $|P|$. L'initialisation à partir de l'enveloppe convexe simplifiée peut permettre d'obtenir directement la surface définitive dans certaines régions, allégeant ainsi le processus de convection géométrique. Cependant, le contrôle de la simplification au cours de la construction de l'enveloppe convexe est peu précis, et le coût total est plus important que celui de l'initialisation avec une boîte englobante. Nous avons donc définitivement adopté une boîte englobante comme surface initiale.

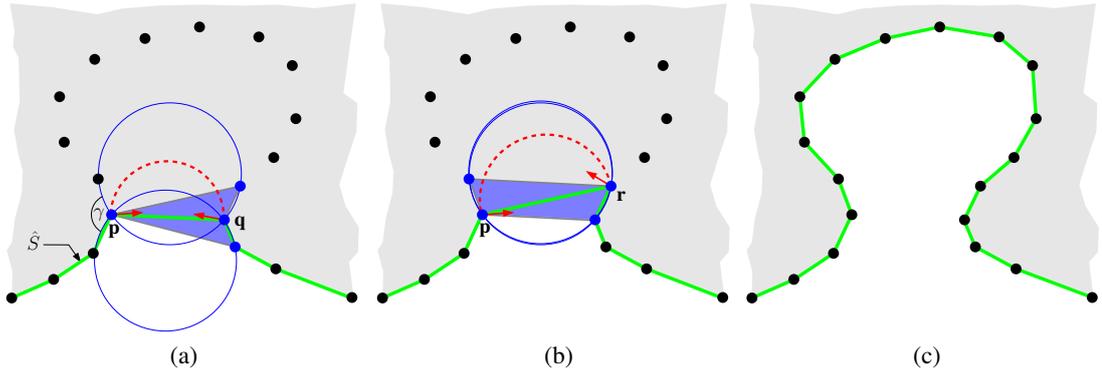


FIG. 4.3 – Détection de poches au cours d'une reconstruction en 2D. En (a), une première poche est détectée. Les normales aux points \mathbf{p} et \mathbf{q} sont presque orthogonales à celle de l'arête \mathbf{pq} et l'angle γ entre les sphères circonscrites aux triangles de Delaunay incidents à \mathbf{pq} est supérieur à $\frac{\pi}{2}$. En (b), une nouvelle poche est détectée. L'angle entre les deux sphères est cette fois proche de π . En (c) apparaît le résultat final de la reconstruction.

4.2 Contrôle de la simplification

Nous calculons le rayon de simplification en un point $\mathbf{p} \in P$ comme une fraction de l'épaisseur locale en ce point :

$$r_{simp}(\mathbf{p}) = \alpha \cdot \tilde{e}(\mathbf{p})$$

L'épaisseur locale discrète contrôle le niveau de redondance toléré dans le voisinage du point \mathbf{p} , déterminé par la valeur du paramètre ρ_{geom} . Le coefficient $\alpha \in [0, 1]$ contrôle la densité de l'échantillon résultant de la reconstruction sélective, et sa distribution à proximité des régions de forte courbure. Nous appelons *facteur d'anticipation* ce facteur, dont l'influence est illustrée par les figures 4.4 et 4.5. Lorsque $\alpha = 1$, on peut observer qu'à proximité des régions de la surface où la courbure change rapidement, la densité de l'échantillon change de manière abrupte, ce qui conduit à la formation de triangles aplatis et de sommets de valence élevée, peu favorables à certains traitements comme la compression par exemple. Choisir une valeur de α inférieure permet d'anticiper sur ces variations dans le processus de sous-échantillonnage, pour obtenir un gradient continu de densité à proximité des régions de forte courbure. L'échantillon résultant devient ainsi presque uniforme localement, et les triangles de forme compacte. Nous avons utilisé la valeur $\alpha = 0.5$ dans tous nos tests, cette valeur fournissant un bon compromis entre densité et qualité des triangles.

La figure 4.6 illustre l'intérêt de combiner les rayons géométrique et topologique dans la mesure d'épaisseur locale sur le modèle SCREWDRIVER. La figure 4.7 illustre l'application de l'algorithme sur le modèle APHRODITE avec différentes valeurs pour le paramètre ρ_{geom} .

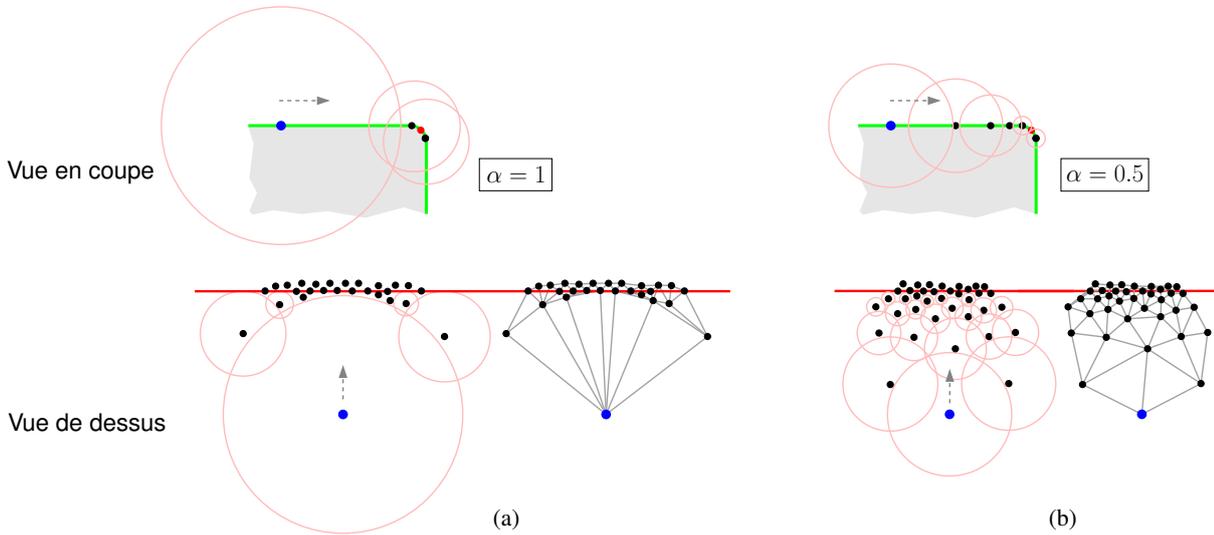


FIG. 4.4 – Effet du facteur d’anticipation sur la distribution des points à proximité d’une arête vive. Les points de l’échantillon sont représentés avec leur région de simplification. En (a), $\alpha = 1$; la densité de l’échantillon après simplification varie de façon importante à proximité de l’arête, produisant des triangles aplatis et des sommets de valence élevée. En (b), $\alpha = 0.5$; la densité d’échantillonnage varie progressivement. Les flèches indique la direction d’avancée de la surface en convection.

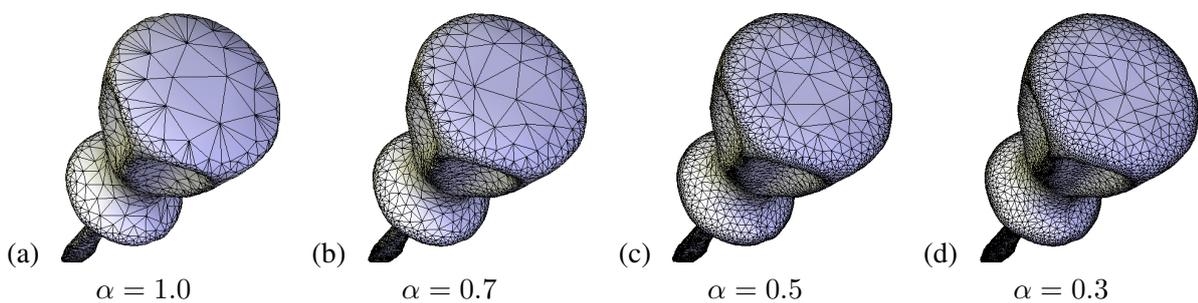


FIG. 4.5 – Résultats de reconstructions sélectives avec des valeurs décroissantes du facteur d’anticipation sur le modèle SCREWDRIVER (ρ_{geom} et ρ_{topo} fixés).

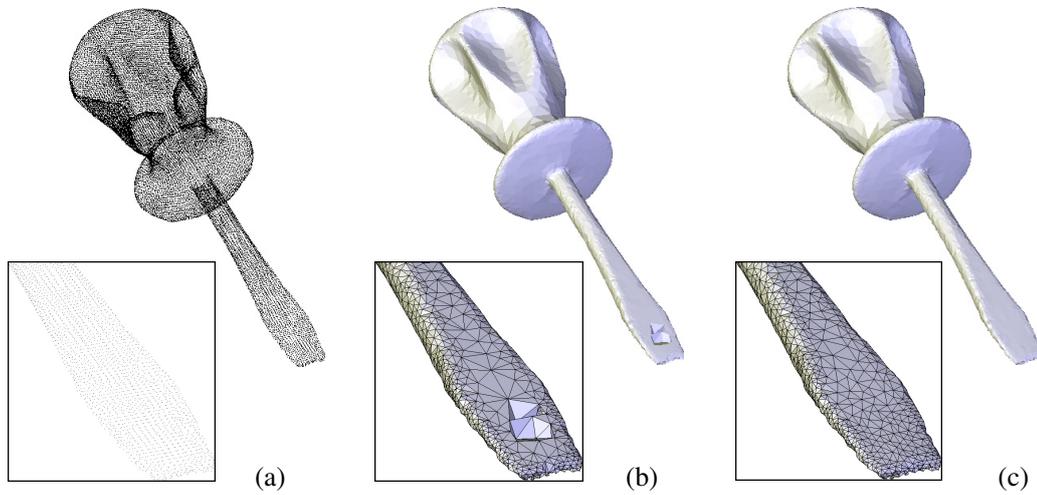


FIG. 4.6 – Reconstruction sélective du modèle SCREWDRIVER. Le nuage de point du modèle original est montré en (a). La reconstruction en (b) a été obtenue en considérant uniquement le rayon géométrique. Elle présente un défaut au niveau de la lame du tournevis dû à une perte d'information topologique. En (c), le modèle a été reconstruit en tenant compte du rayon géométrique et du rayon topologique dans l'estimation de l'épaisseur locale.

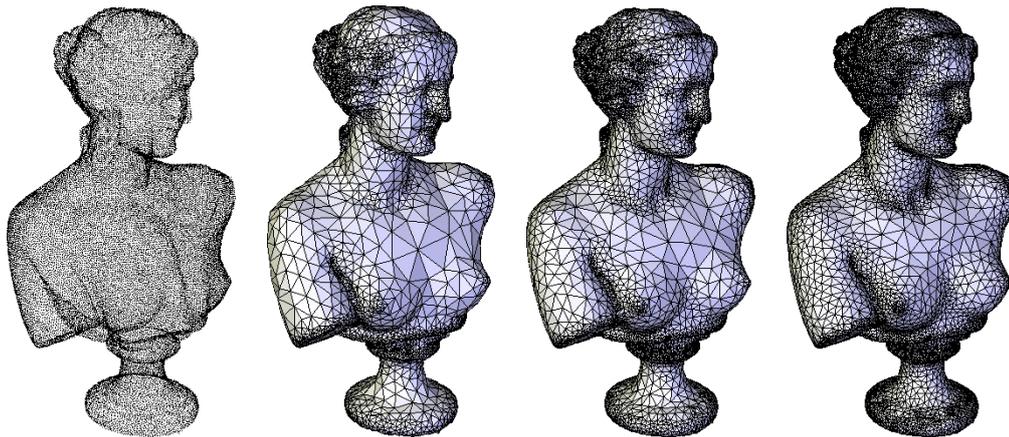


FIG. 4.7 – Reconstructions du modèle APHRODITE (à gauche, 46k points) avec $\alpha = 0.5$ et différentes valeurs de ρ_{geom} , de gauche à droite : 0.65 (8.0k points), 0.8 (12.2k points) et 0.9 (17.8k points).

4.3 Implantation efficace

Dans cette section, nous nous intéressons à la structure de données et aux algorithmes à mettre en œuvre pour implanter l’algorithme de reconstruction sélective de manière efficace.

4.3.1 Position du problème

Comme beaucoup de points peuvent être supprimés par l’algorithme de reconstruction sélective, construire la triangulation de Delaunay de tous les points peut être inutilement coûteux. Dans ce cas, d’autres prédicats et requêtes sont nécessaires pour permettre à la surface d’évoluer. L’évaluation de \mathbf{P}_{go} implique une requête supplémentaire \mathbf{Q}_{db} qui retourne les points de $P \setminus \hat{R}$ situés dans la demi-sphère de Gabriel associée à la facette. La requête \mathbf{Q}_{dt} nécessite aussi un prédicat \mathbf{P}_{ct} permettant de vérifier si un point dans $P \setminus \hat{R}$ entre en conflit avec un tétraèdre.

Il s’agit maintenant de déterminer une structure de données appropriée pour répondre efficacement à \mathbf{P}_{go} , \mathbf{Q}_{dt} et \mathbf{Q}_{pv} . Une alternative à la triangulation de Delaunay consiste à recourir à une structure de localisation spatiale de type *kd*-arbre. Une structure de *kd*-arbre est adaptée à la recherche des plus proches voisins d’un point de l’échantillon de manière incrémentale, mais elle peut être lourde à mettre en œuvre pour répondre au prédicat \mathbf{P}_{go} , et n’est clairement pas adaptée aux requêtes \mathbf{Q}_{dt} . En effet, chaque évaluation du prédicat \mathbf{P}_{go} nécessite alors d’effectuer la requête \mathbf{Q}_{db} sur les données. Or, les demi-sphères de Gabriel peuvent contenir une grande partie de l’échantillon de points en entrée, en particulier au début du processus de reconstruction, qui doivent ensuite être tous testés pour répondre à la requête \mathbf{Q}_{dt} . Selon la géométrie de la surface, le nombre de points contenu dans les demi-sphères décroît plus ou moins rapidement (fig. 4.8), ce qui affecte aussi la complexité de l’algorithme. En outre, quand \mathbf{P}_{go} est satisfait et qu’une poche est détectée, l’espace de recherche pour \mathbf{Q}_{dt} s’étend à tout le demi-espace au-dessus du plan de la facette. La difficulté pour répondre à \mathbf{P}_{go} et à \mathbf{Q}_{dt} est en fait liée un manque de connaissances sur le domaine inexploré borné par la surface en convection. Il apparaît donc nécessaire de structurer ce domaine afin de réduire la taille des espaces de requête.

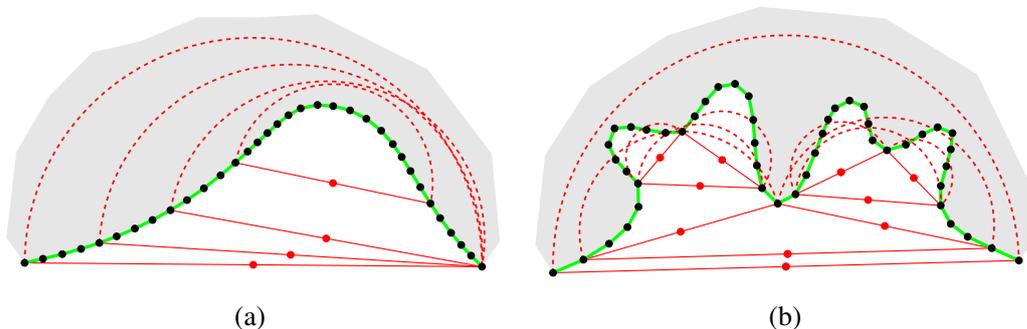


FIG. 4.8 – Illustration en 2D de la décroissance du nombre de points contenu dans les demi-sphères de Gabriel en fonction de la géométrie locale. En (a), le nombre de points décroît de manière linéaire lors de la traversée de la concavité. En (b), alors que plusieurs concavités sont successivement traversées, le nombre de points décroît de manière polynomiale.

4.3.2 Structure de données

Notre solution consiste à combiner un kd -arbre avec la triangulation de Delaunay des points retenus dans la surface en convection. Nous enrichissons la partie inexplorée de cette triangulation avec des points *balises* qui contribuent à structurer ce domaine.

Considérons la triangulation de Delaunay des points retenus à une étape donnée du processus de reconstruction sélective. Chaque facette de la surface en convection est l'interface entre deux cellules ; nous appelons *cellule frontale* celle qui est orientée vers l'intérieur de la surface. Les cellules frontales sont connectées à des sommets opposés sur la surface et peuvent fournir des informations sur l'étendue du domaine inexploré. Cependant, la partie de leur sphère circonscrite située à l'intérieur de la surface peut recouvrir un domaine spatial plus grand que les demi-sphères de Gabriel. La triangulation de Delaunay des points retenus est par conséquent insuffisante pour réduire l'espace de recherche pour les requêtes géométriques impliquées dans l'algorithme de convection géométrique : des points supplémentaires sont nécessaires pour « casser » les grandes cellules frontales.

Nous partons d'un ensemble de points balises obtenus à partir d'une structure de kd -arbre avec un seuil sur le nombre maximum de points par feuille. Dans chaque cellule feuille, le point qui est le plus proche du centre de la cellule est retenu comme un point balise. La triangulation de Delaunay \hat{D} de ces points est ensuite construite, et enrichie avec les sommets d'une boîte englobante alignée selon les axes de coordonnées. La surface est initialisée sur l'enveloppe convexe, i.e. à partir de la boîte englobante, et est directement supportée par la triangulation de Delaunay \hat{D} . Le processus de reconstruction peut être alors lancé en bénéficiant de cellules frontales plus petites qui vont aider à accélérer l'évaluation de \mathbf{P}_{go} et \mathbf{Q}_{dt} . En parallèle, les recherches spatiales sont déléguées à une structure de kd -arbre qui stocke l'ensemble de points, avec un algorithme spécifique qui sera décrit par la suite.

4.3.3 Algorithme de reconstruction accéléré

L'algorithme accéléré met à jour dynamiquement la triangulation \hat{D} au cours du processus de reconstruction en insérant les points retenus et en retirant les points balises non retenus de manière à ce que ces derniers n'affectent pas le résultats. Nous exploitons ici la propriété que les cellules de Delaunay qui deviennent externes à la surface demeurent jusqu'à la fin du processus, ce qui n'est pas le cas des cellules internes. Les cellules externes sont naturellement protégées contre les insertions ou les retraits de sommets suivants.

Nous poursuivons avec les notations introduites précédemment pour décrire l'algorithme accéléré. La figure 4.9 illustre l'algorithme sur un échantillon de points en deux dimensions. Pour vérifier si une facette orientée \mathbf{pqr} de la surface \hat{S} vérifie \mathbf{P}_{go} , nous considérons sa cellule frontale σ dans la triangulation de Delaunay \hat{D} courante ; sa sphère circonscrite est notée \mathcal{S} . Nous appelons *sommet frontal* le sommet de σ opposé à la facette \mathbf{pqr} ; sa position sera notée \mathbf{s} . La demi-sphère de Gabriel de la facette est finalement notée \mathcal{B} , et le demi-espace au-dessus du plan qui la supporte est noté \mathcal{H} . La première étape pour évaluer \mathbf{P}_{go} consiste à vérifier si \mathbf{s} se trouve à l'intérieur ou à l'extérieur de \mathcal{B} .

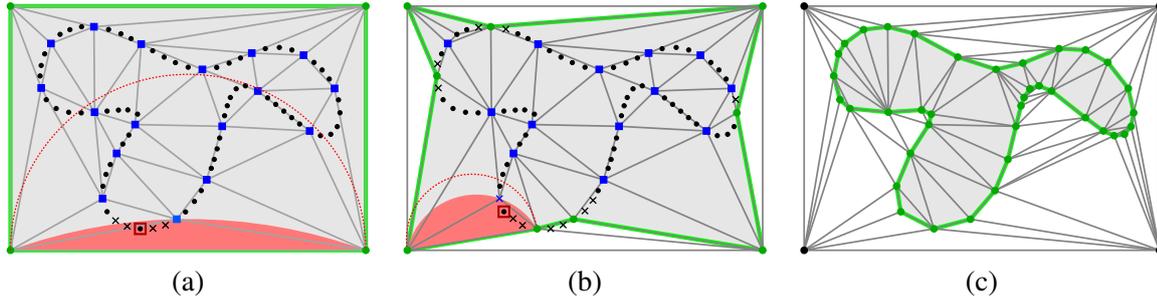


FIG. 4.9 – Reconstruction sélective accélérée sur un échantillon de points 2D. En (a), une courbe englobant l'échantillon de points est initialisée sur une boîte englobante. Une triangulation de Delaunay a été construite à partir de ses sommets et d'un ensemble de balises (points bleus). L'arête courante a une demi-sphère de Gabriel non vide : le sommet frontal dans la triangulation se trouve à l'intérieur. Le point qui forme un triangle de Delaunay avec l'arête est recherché dans le disque circonscrit à la cellule frontale de la triangulation (le point entouré d'un carré). L'ensemble de points est alors simplifié localement autour du point retenu (les points en forme de croix sont éliminés). En (b), le point retenu a été inséré dans la triangulation et la courbe poursuit son évolution. Les facettes attachées aux sommets de la boîte englobante sont forcées. Le résultat final est présenté en (c).

- Si $s \in \mathcal{B}$, alors le prédicat \mathbf{P}_{go} n'est pas satisfait. La requête \mathbf{Q}_{dt} est alors effectuée dans $(P \setminus \hat{R}) \cap \mathcal{S} \cap \mathcal{H}$, qui correspond à l'ensemble des points qui entrent en conflit avec σ . Si cet ensemble est vide, alors \mathbf{pqrs} forme un tétraèdre de Delaunay dans $P \setminus \hat{R}$.
- Si $s \notin \mathcal{B}$, il n'est pas garanti que \mathbf{P}_{go} soit satisfait. Pour évaluer ce prédicat, nous récupérons d'abord tous les points dans l'ensemble $(P \setminus \hat{R}) \cap \mathcal{B}$ par \mathbf{Q}_{db} . Si cet ensemble n'est pas vide, alors \mathbf{P}_{go} n'est pas satisfait et \mathbf{Q}_{dt} est ensuite effectué dans l'ensemble $(P \setminus \hat{R}) \cap \mathcal{B}$.

Dans le cas où \mathbf{P}_{go} est satisfait mais qu'une poche est détectée, alors \mathbf{Q}_{dt} est effectuée dans l'ensemble de points $(P \setminus \hat{R}) \cap \mathcal{S} \cap \mathcal{H}$ qui entrent en conflit avec σ . Si cet ensemble est vide, alors \mathbf{pqrs} forme un tétraèdre de Delaunay dans $P \setminus \hat{R}$.

À chaque fois qu'un nouveau tétraèdre de Delaunay est formé à partir d'une facette \mathbf{pqr} et d'un point \mathbf{x} , alors \mathbf{x} est inséré dans la triangulation de Delaunay \hat{D} à condition que $\mathbf{x} \neq \mathbf{s}$, puis la surface est mise à jour. Notons que toute facette attachée à des sommets de la boîte englobante doit être ouverte, c'est-à-dire que la requête \mathbf{Q}_{dt} doit être effectuée, même quand le prédicat \mathbf{P}_{go} est satisfait.

Nous discutons maintenant du choix des points balises. Le principal bénéfice de ces points est au début du processus, alors que les demi-sphères de Gabriel peuvent contenir un grand nombre de points de l'échantillon. À mesure que leur taille décroît, ce bénéfice diminue aussi, parce que la densité de ces points devient insuffisante. Cependant, les petites demi-sphères de Gabriel peuvent être traitées de manière efficace. Si cette densité est trop grande, alors beaucoup de temps peut être passé à retirer des points balises indésirables. Comme le taux de simplification final dépend beaucoup de la forme de la valeur de ρ_{geom} , le nombre optimal de points balises n'est pas facile à déterminer. En pratique, choisir un point balise pour quelques milliers de points (entre 1k et 10k) est suffisant pour limiter les sphères circonscrites aux cellules frontales à quelques centaines de points dans le pire cas, et obtenir une accélération significative du processus de reconstruction.

4.3.4 Recherche spatiale accélérée

Dans l'algorithme de reconstruction sélective accéléré, le prédicat $\mathbf{P}_{\mathbf{go}}$ est évalué en localisant d'abord des sous-ensembles de points qui entrent en conflit avec les cellules frontales ou qui se trouvent à l'intérieur des demi-sphères de Gabriel. Quand l'ensemble de points retourné n'est pas vide pour une facette donnée, la requête $\mathbf{Q}_{\mathbf{dt}}$ est effectuée, i.e. il s'agit de trouver le point formant un tétraèdre de Delaunay avec la facette. Sans information sur la structure de l'échantillon de points, tout point de cet ensemble est un candidat potentiel, et doit ainsi être testé. Pour réduire le nombre de tests, nous les ordonnons en nous appuyant sur une structure de kd -arbre.

Nous examinons d'abord le cas simple dans lequel une facette de la surface est telle que son sommet de Delaunay frontal est localisé dans la demi-sphère de Gabriel de la facette. Nous commençons par chercher les feuilles non vides du kd -arbre susceptibles de contenir les points qui se trouvent à l'intérieur de la sphère circonscrite de la cellule frontale, restreinte au demi-espace au-dessus du plan de la facette ; nous appelons \mathcal{C} cette région. Cette étape est réalisée par un parcours en profondeur du kd -arbre. Si une cellule du kd -arbre se trouve complètement à l'intérieur de \mathcal{C} , alors les feuilles du sous-arbre correspondant sont retournées. Les feuilles qui n'intersectent \mathcal{C} que partiellement sont aussi retournées. Tester si une cellule du kd -arbre intersecte \mathcal{C} implique deux prédicats : un test de recouvrement sphère/boîte et un test de recouvrement demi-espace/boîte [146]. Un compteur donnant le nombre de points restants dans une feuille permet d'éviter de tester les feuilles vides.

Quand des feuilles non vides sont retournées, l'étape suivante consiste à obtenir le point qui forme un tétraèdre de Delaunay avec la facette, avec une complexité moyenne meilleure que linéaire par rapport au nombre de points contenus dans les feuilles. Notre algorithme procède de manière incrémentale, en commençant avec le point associé au sommet frontal comme candidat. Les feuilles obtenues pour la facette sont stockées dans une file notée L et la facette est notée \mathbf{pqr} . La procédure est résumée par l'algorithme 2 ci-dessous.

Algorithme 2 $\mathbf{Q}_{\mathbf{dt}}$ (facette \mathbf{pqr})

t'arque L contient plus d'un élément **faire**

- Récupérer un point dans chaque feuille de kd -arbre de L qui se trouve dans \mathcal{C} , s'il en existe. Soit M cet ensemble de points ;
- Rechercher dans M le meilleur candidat \mathbf{c} , i.e. le point tel que la sphère circonscrite de \mathbf{pqrc} ne contienne aucun autre point de M , en utilisant le prédicat $\mathbf{P}_{\mathbf{ct}}$;
- Retirer de L les cellules vides et les cellules qui n'entrent pas en conflit avec le tétraèdre \mathbf{pqrc} ;

fin tantque

Retourner le meilleur candidat parmi les points restants ;

Le cas où la facette a son sommet frontal à l'extérieur de la demi-sphère de Gabriel est traité de manière similaire, excepté que les conflits sont d'abord testés dans l'ensemble des feuilles qui intersectent la demi-sphère de Gabriel pour déterminer si la facette satisfait $\mathbf{P}_{\mathbf{go}}$. Dès que le prédicat est n'est pas vérifié ou si une poche est détectée, alors la recherche est poursuivie pour répondre à la requête $\mathbf{Q}_{\mathbf{dt}}$.

La méthode permet d'éliminer rapidement les feuilles du kd -arbre qui sont le moins susceptibles de

contenir le bon candidat. Cependant, il est souvent difficile de décider entre les feuilles restantes, étant donné que les candidats peuvent « sauter » d’une cellule feuille à une autre. Quand le nombre de feuilles restantes se stabilise, nous arrêtons la procédure et passons à une recherche linéaire parmi les points qui restent de manière à éviter tout surcoût lié aux tests de conflits entre les cellules feuilles et les cellules de la triangulation. En pratique, le gain par facette est typiquement 10% à 20% de tests \mathbf{P}_{ct} de conflit entre un point et un tétraèdre.

4.4 Résultats expérimentaux

Nous avons implanté notre méthode de reconstruction sélective sous Linux à l’aide de la bibliothèque CGAL [1] (Computational Geometry Algorithm Library). Nous avons recours à CGAL pour la construction et la mise à jour des triangulations de Delaunay et la construction des k d-arbres. Nous utilisons les prédicats filtrés pour la vérification robuste de la propriété de Gabriel orientée pour une facette (\mathbf{P}_{go}) et du critère de Delaunay pour un tétraèdre (\mathbf{P}_{ct}).

Les figures 4.10, 4.11, 4.12, 4.13, 4.14, 4.15 et 4.16 montrent des résultats de reconstructions sélectives sur divers échantillons de points. Dans les paragraphes suivants, nous commençons par discuter des divers types de données manipulées avant de nous intéresser à la qualité des maillages produits et aux performances de la méthode.

4.4.1 Données testées et spécificités de leur traitement

Nous avons appliqué notre méthode à trois types de données :

- des échantillons de points issus de la fusion d’images de profondeur recalées par des méthodes telles que celles de Curless et Levoy [67] ou de Rocchini et al. [164] ;
- des échantillons de points provenant directement de l’union d’images de profondeur recalées ;
- des échantillons de points provenant de l’union d’images de profondeur avec position et couleur de chaque point l’espace RVB.

Les données du premier type correspondent à l’échantillonnage d’une surface implicite approximant les données des images de profondeur sur une grille régulière avec une résolution proche de celle des images. Dans les autres cas, les données sont *brutes*, c’est-à-dire qu’elles n’ont pas subi de prétraitement. Les données avec couleur nous ont été fournies par l’équipe du LIRMM impliquée dans le projet Art3D.

Images de profondeur fusionnées Des imprécisions dans le recalage d’un ensemble d’images de profondeur sont sources de bruit dans les données. Les techniques de fusion d’images de profondeurs sont utiles pour résorber les écarts entre les images ainsi que pour compléter les régions où il manque des données, par diffusion volumétrique. Ces données sont en général très denses et de bonne qualité, même si la fusion n’élimine pas totalement le bruit (fig. 4.11, 4.13, 4.12, 4.14).

Données brutes L'algorithme de convection géométrique est robuste au bruit de type gaussien qui peut être présent dans des données brutes (fig. 4.10(a)). Dans le cadre de la reconstruction sélective, la procédure de simplification nécessite de lisser suffisamment le champ des normales estimées pour donner un taux de simplification satisfaisant. Dans ce cas, nous estimons la direction des vecteurs normaux localement en considérant des voisinages relativement larges (fig. 4.10(b)), ce qui produit l'effet d'un filtre passe-bas sur le champ des normales.

L'algorithme de convection géométrique a aussi la capacité de boucher certains trous, dans la mesure où les facettes qui les ferment sont considérées comme définitives. Dans les autres cas de surfaces échantillonnées à bords, la surface reconstruite forme une enveloppe autour de l'échantillon de points qui doit alors être « dédoublée » (fig. 4.15). Il peut être parfois intéressant aussi de bloquer interactivement les facettes attachées à des bords.

Données avec couleur Un échantillon de points peut être fourni avec un certain nombre d'attributs scalaires ou vectoriels, tels qu'une couleur en chaque point. La couleur peut notamment être intéressante pour restituer l'apparence visuelle des objets numérisés. En simplifiant les données, il est alors important d'en tenir compte afin d'obtenir un maillage dans lequel les discontinuités entre les couleurs sont préservées.

La mesure de redondance géométrique impliquée dans la procédure de simplification de l'algorithme de reconstruction sélective peut être facilement remplacée ou complétée par des critères portant sur des attributs définis en chaque point. Pour les intégrer au processus, il suffit de disposer d'une mesure de similarité entre deux valeurs d'un attribut donné. Dans le cas particulier de la couleur, nous avons utilisé le critère suivant. Nous supposons qu'à tout point \mathbf{q} de l'échantillon P est associé un triplet $\mathbf{q}^c = (r, v, b) \in [0, 1]^3$ où r , v et b désignent les composantes rouge, vert et bleu. Un point \mathbf{p}_i voisin de $\mathbf{p} \in \hat{S}$ est considéré comme colorimétriquement redondant par rapport à \mathbf{p} si la condition suivante est vérifiée :

$$(C_{coul}) \quad \frac{1}{\sqrt{3}} \|\mathbf{p}_i^c - \mathbf{p}^c\| \leq \rho_{coul} \quad \rho_{coul} \in [0, 1]$$

Ce critère peut être utilisé en complément de la mesure d'épaisseur locale pour contraindre davantage la simplification en sur-échantillonnant les régions lisses présentant des discontinuités de couleur liées à des motifs. Par ce moyen, il est possible de reconstruire directement une surface triangulée simplifiée adaptée à une visualisation par lissage de Gouraud ou à la construction de cartes de textures avec une perte d'information contrôlée sur la couleur.

Les figures 4.15 et 4.16 montrent des résultats de reconstruction sélective obtenus en appliquant le critère C_{coul} . De nombreuses mesures de similarité entre couleurs ont été développées dans le domaine de l'indexation d'images. D'autres espaces de couleur ou d'autres mesures d'erreur que la distance euclidienne pourraient être considérées, comme par exemple tenant compte de la perception humaine [166].

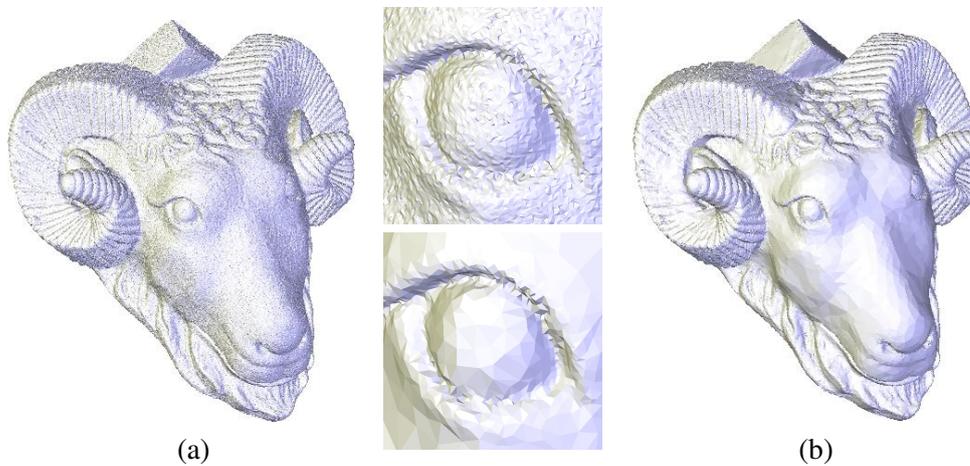


FIG. 4.10 – Reconstruction du modèle RAM (622k points); (a) : reconstruction du modèle original sans simplification ; (b) : résultat d’une reconstruction sélective avec $\rho_{geom} = 0.8$ (94% de points éliminés). En ayant estimé les normales en chaque point en considérant les 18 plus proches voisins, le rayon géométrique impliqué dans notre mesure d’épaisseur locale se comporte comme un filtre sur le bruit.

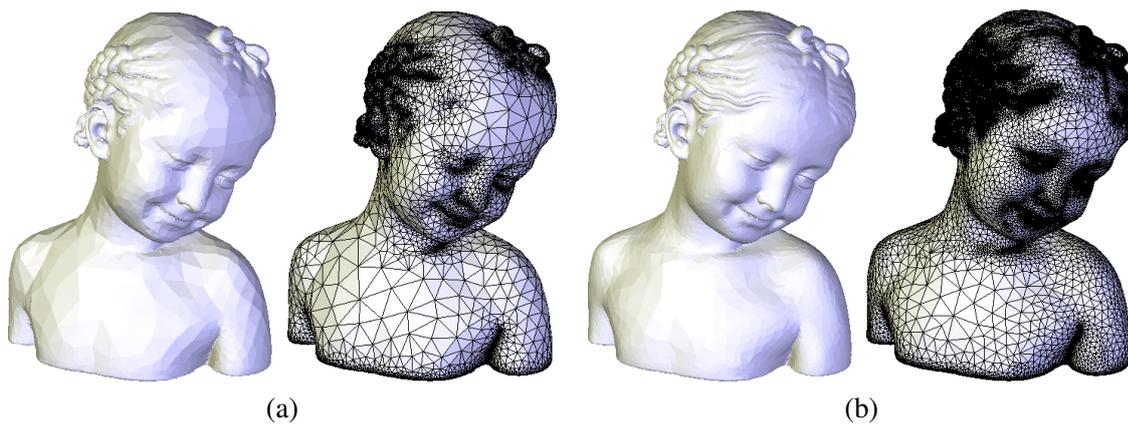


FIG. 4.11 – Résultats de reconstructions sélectives sur le modèle BIMBA (1.9M points); (a) : reconstruction obtenue avec $\rho_{geom} = 0.65$ (98% de points éliminés); (b) : reconstruction obtenue avec $\rho_{geom} = 0.9$ (95% de points éliminés).

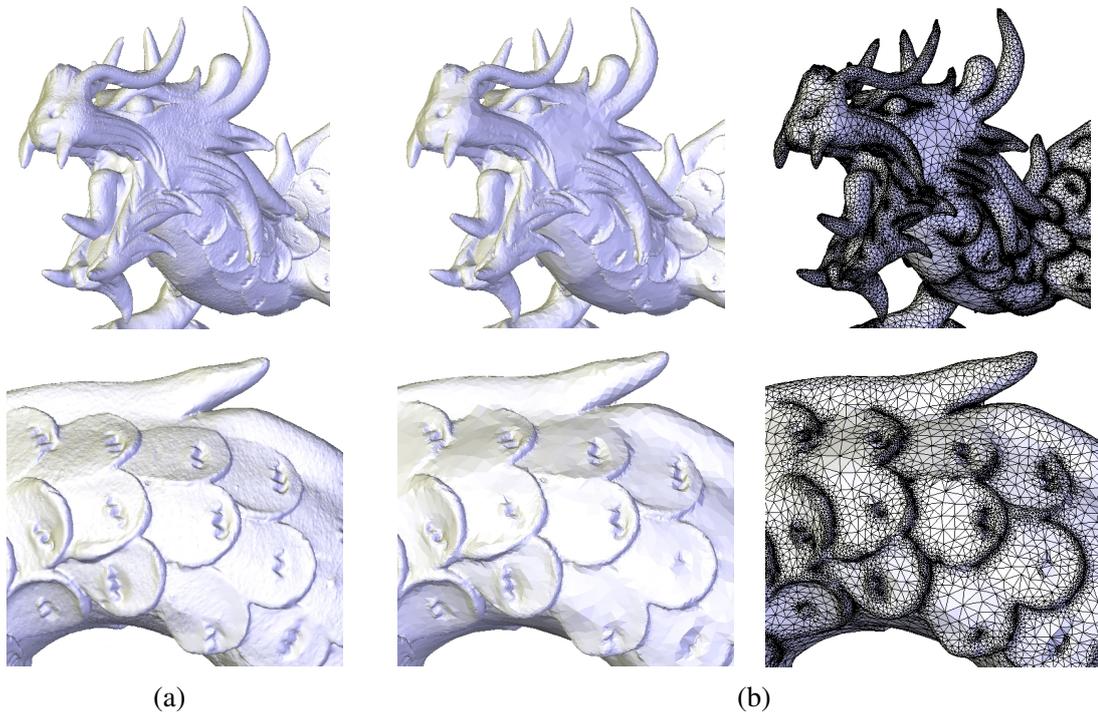


FIG. 4.12 – Reconstruction du modèle DRAGON ; (a) : reconstruction du modèle original ; (b) : résultat d'une reconstruction sélective avec $\rho_{geom} = 0.65$ (95% de points éliminés).

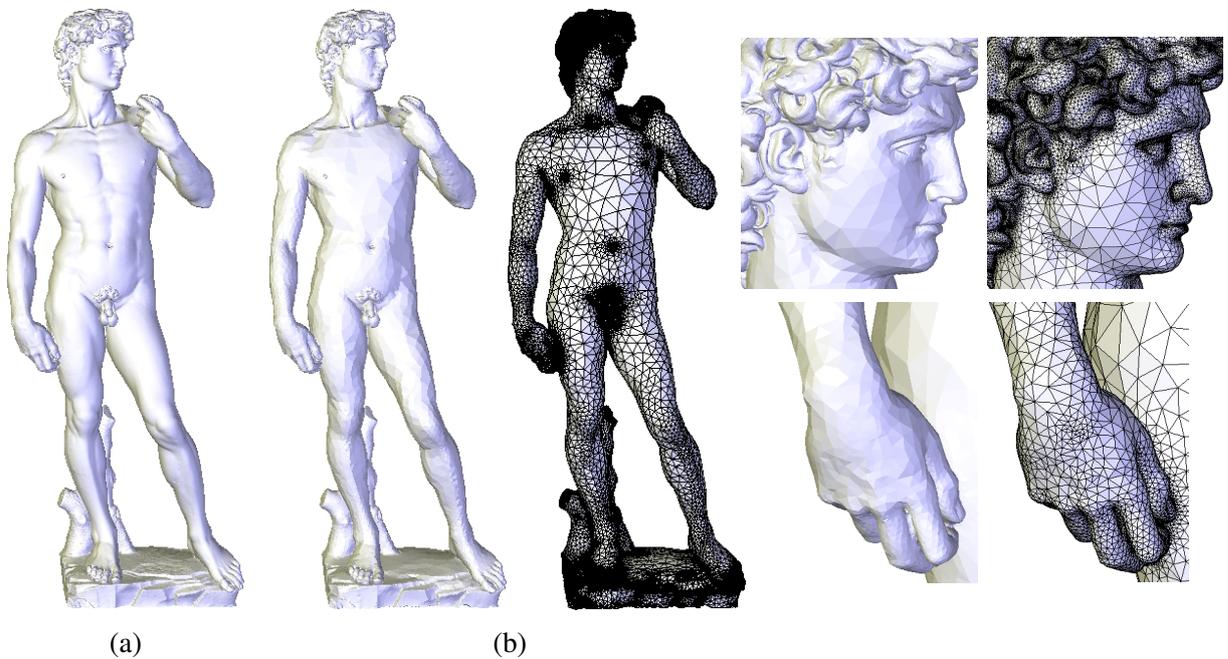


FIG. 4.13 – Reconstruction du modèle DAVID (3.6M points) ; (a) : reconstruction du modèle original ; (b) : résultat d'une reconstruction sélective avec $\rho_{geom} = 0.65$ (96% de points éliminés).

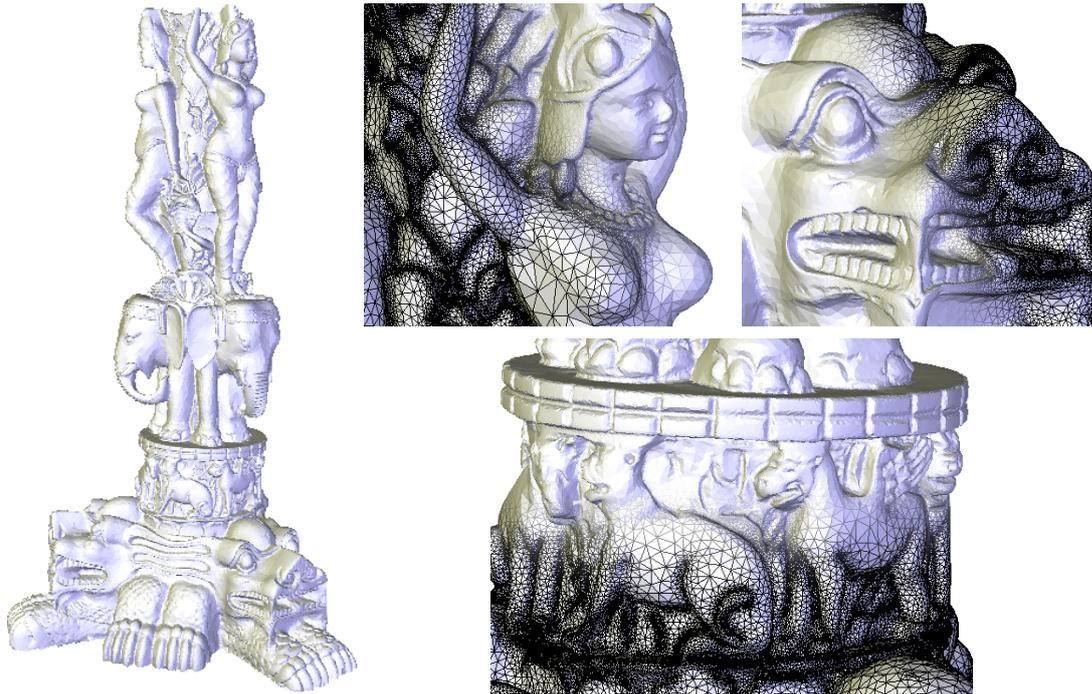


FIG. 4.14 – Résultat d’une reconstruction sélective du modèle THAI STATUE (5M points) avec $\rho_{geom} = 0.65$ (89% de points éliminés).

4.4.2 Qualité des maillages

Tous les modèles ont été reconstruits avec un facteur d’anticipation de 0.5. Nos modèles simplifiés comportent typiquement de 95% à 99% de sommets de moins que le nombre de points de l’échantillon original, ce qui suffit souvent à préserver la forme des objets numérisés à un niveau d’échelle moyen, et même à un niveau fin si l’échantillon de points est très redondant. La méthode est capable de produire des modèles maillés simplifiés de bonne qualité directement, sans qu’il soit nécessaire d’améliorer la qualité du maillage par la suite. La majorité des sommets ont une valence entre 5 et 7, et la plupart des facettes ont une forme compacte (fig. 4.17).

Des triangles presque équilatéraux et des sommets de valence proche de 6 sont favorables pour certaines applications, comme la compression ou les méthodes d’éléments finis. Pour mesurer la qualité des maillages obtenus, nous calculons d’une part la distribution des valences des sommets et d’autre part la qualité de la forme des triangle en utilisant la mesure de compacité d’un triangle proposée par Guézic [105] :

$$c = \frac{4\sqrt{3}a}{l_1^2 + l_2^2 + l_3^2}$$

où les l_i sont les longueurs des arêtes d’un triangle et a est l’aire du triangle. Cette mesure vaut 1 pour un triangle équilatéral et 0 pour un triangle dont les sommets sont colinéaires.

Nous n’avons pas évalué la qualité de l’approximation fournie par notre méthode par rapport aux données originales, qui serait intéressante dans le cadre d’une comparaison avec d’autres techniques de

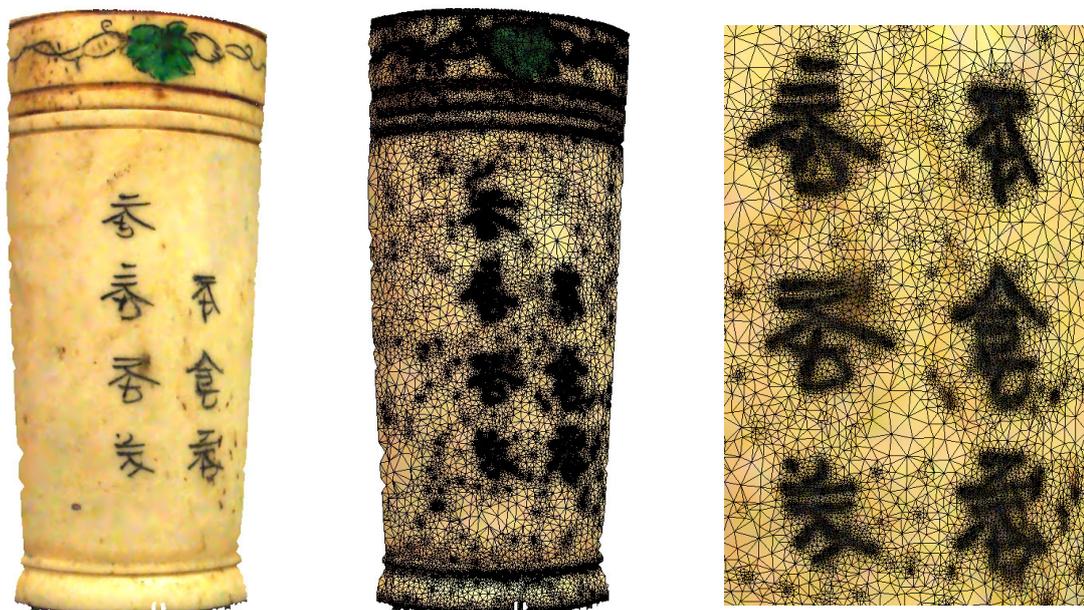


FIG. 4.15 – Reconstruction sélective du modèle IVOIRE (géométrie et couleur). Le modèle original comporte 311k points. Une surface triangulée a été reconstruite en combinant la mesure d'épaisseur locale et le critère C_{coul} avec $\rho_{geom} = 0.65$ et $\rho_{coul} = 0.1$, éliminant 76% des points.

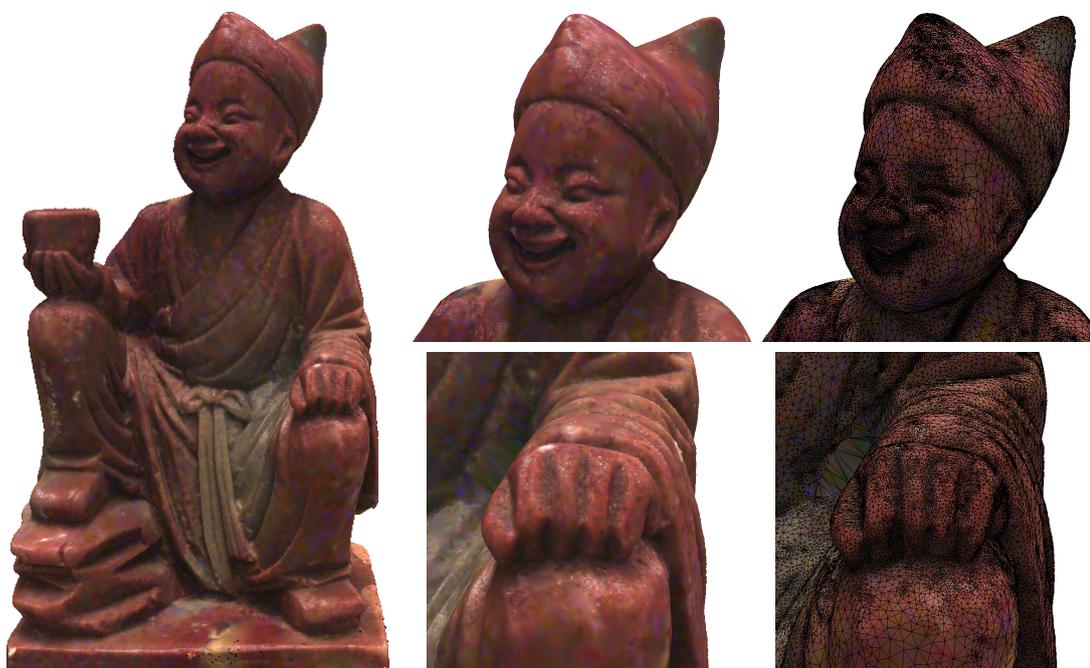


FIG. 4.16 – Reconstruction sélective du modèle SAGE (5M points avec couleur) avec $\rho_{geom} = 0.65$ et $\rho_{coul} = 0.1$ (97% de points éliminés).

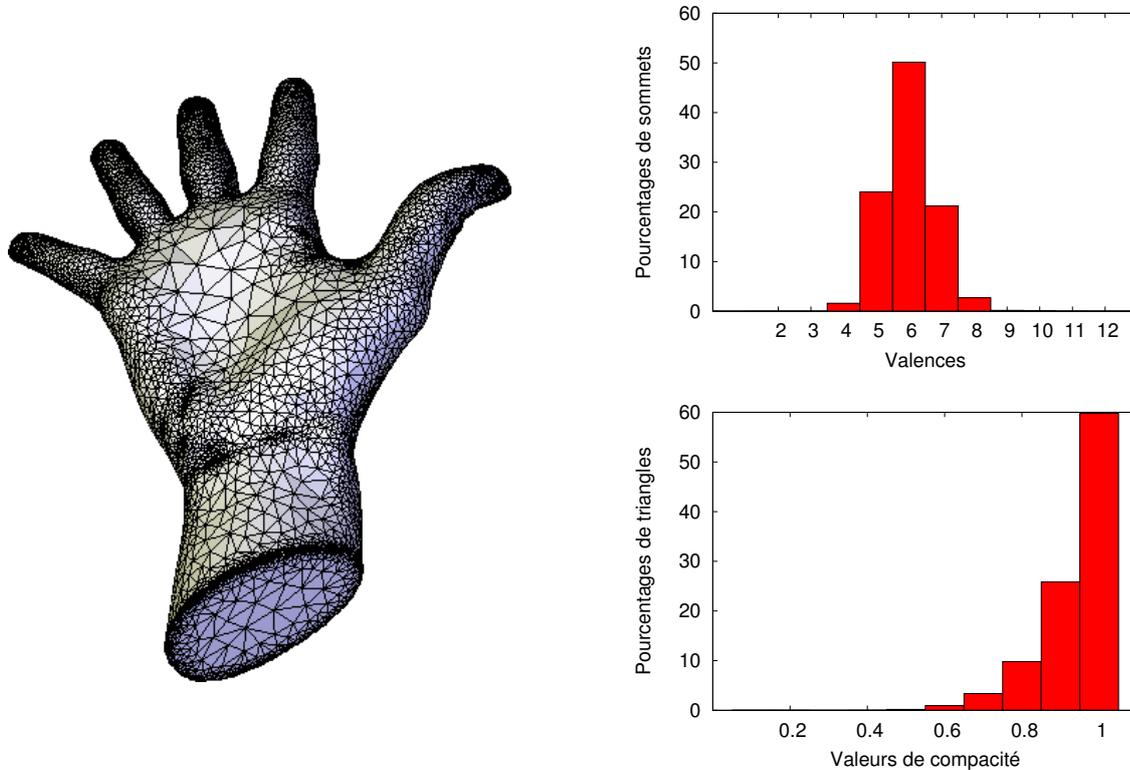


FIG. 4.17 – Évaluation de la qualité des maillages produits par notre méthode de reconstruction sélective par les valeurs de valence des sommets et de compacité des triangles.

simplification d'échantillons de points ou de simplification de maillage. On peut toutefois souligner que le critère de simplification mis en œuvre, en particulier celui qui contrôle le rayon géométrique dans le calcul de l'épaisseur locale, a tendance à lisser la surface dans les régions de forte courbure. Or, il serait intéressant de parvenir à conserver les arêtes vives intactes au cours de la simplification. Une mesure d'erreur géométrique plus adaptée pourrait être celle proposée par Ohtake et al. [156], combinée à notre rayon topologique, mais la reconstruction des arêtes vives implique un rééchantillonnage, qui n'est encore pas pris en charge par notre méthode.

4.4.3 Performances

Les temps de calcul et l'utilisation de la mémoire sont répertoriés dans la table 4.1. Tous les résultats présentés ici ont été obtenus sur un processus Pentium IV 3.0 GHz avec 2 Go de RAM. Ces temps de calcul incluent le temps de prétraitement pour construire la structure de kd-arbre, sélectionner les points balises et construire la triangulation de Delaunay initiale. La table 4.2 présente le profil d'exécution

global pour l'algorithme de reconstruction sélective. Pour les reconstructions sélectives avec toutes les données chargées en mémoire, nous avons fixé le nombre points balises à 1 pour 2000 points de l'échantillon. Le temps de prétraitement est inférieur à 12 secondes dans tous nos tests.

Les profils d'exécution (table 4.2) montrent qu'évaluer \mathbf{P}_{go} et \mathbf{Q}_{dt} est de loin la tâche la plus coûteuse dans l'algorithme de reconstruction sélective. Alors que nous avons réduit les espaces de requête, le coût global des recherches spatiales demeure proportionnel au nombre de facettes à travers lesquelles la surface passe, ce qui représente la principale limite de la méthode actuelle en termes de performances. Le coût en mémoire est aussi relativement élevé à cause du stockage de l'échantillon de points, de leur organisation dans un k d-arbre, et d'une triangulation de Delaunay partielle.

Modèle		Reconstruction sélective			
Nom	$ P $	ρ_{geom}	$ P_{simp} $	Temps	Mémoire
TRIPLE HÉCATE	90 180	0.85	30 832	0 :19	43 Mo
APHRODITE	46 096	0.65	8 054	0 :06	23 Mo
		0.8	12 259	0 :09	26 Mo
		0.9	17 895	0 :14	29 Mo
RAM	622 716	0.9	93 646	0 :52	178 Mo
		0.8	51 342	0 :42	156 Mo
BIMBA	1 873 832	0.65	31 643	0 :47	380 Mo
		0.9	101 573	1 :24	395 Mo
ASIAN DRAGON	3 609 600	0.65	185 504	2 :48	778 Mo
DAVID	3 617 008	0.6	137 025	2 :06	754 Mo
THAI STATUE	5 001 964	0.65	571 600	6 :21	1367 Mo
IVOIRE (couleur)	310 806	0.65(0.1)	75 032	1 :15	162 Mo
SAGE (couleur)	5 049 913	0.65(0.1)	161 418	6 :13	1214 Mo

TAB. 4.1 – Performances de notre méthode de reconstruction sélective pour différents échantillons de points. Les temps de calcul sont donnés en minutes:secondes et incluent les temps de prétraitement (construction du k d-arbre et de la triangulation de Delaunay initiale). La colonne de « mémoire » correspond à la quantité maximum de mémoire utilisée durant les reconstructions, en mégaoctets. Tous les tests ont été effectués sur un processeur Pentium IV 3.0 GHz avec 2 Go de RAM.

Modèle	BIMBA	ASIAN DRAGON	THAI STATUE
ρ_{geom}	0.9	0.65	0.65
Prétraitements	8.7	6.5	4.2
Évaluation de \mathbf{P}_{go} et \mathbf{Q}_{dt}	54.9	66.8	74.0
Évaluation de \mathbf{Q}_{pv}	30.7	20.2	13.3
Retrait/insertion de sommets	5.7	6.5	8.5

TAB. 4.2 – Profil d'exécution pour trois reconstructions sélectives. Pour chaque modèle, la colonne indique les pourcentages du temps global d'exécution passé pour accomplir les différentes tâches répertoriées dans la colonne de gauche.

Mise à jour dynamique

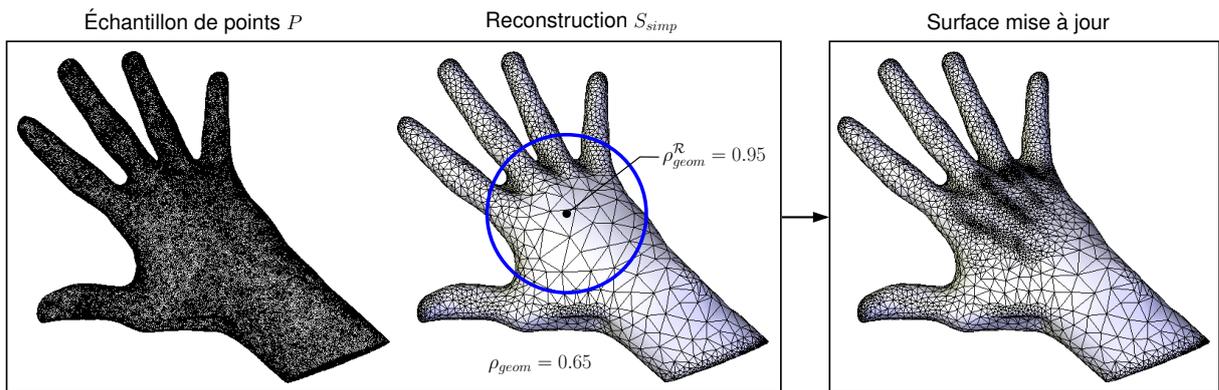


FIG. 5.1 – Application de la mise à jour dynamique à la correction interactive d'un modèle reconstruit de manière sélective.

Dans ce deuxième chapitre, nous présentons un algorithme permettant de mettre à jour dynamiquement le résultat d'une reconstruction par convection géométrique, sélective ou non. Nous commençons par décrire l'algorithme général de mise à jour locale, avant de présenter une application de correction interactive du niveau de précision de modèles reconstruits de manière sélective.

5.1 Algorithme de mise à jour locale

Rappelons qu'une (pseudo-)surface \hat{S} qui interpole un sous-ensemble \hat{P} d'un échantillon de points P est un sous-ensemble de facettes de la triangulation de Delaunay \hat{D} de \hat{P} . A la fin du processus de convection géométrique, toutes les facettes de \hat{S} orientées vers l'intérieur vérifient la propriété de Gabriel orientée par rapport à \hat{P} . Quand un point \mathbf{p} est inséré (resp. retiré) dans la triangulation de Delaunay, cette dernière est modifiée *localement* dans une région composée d'un ensemble connexe de cellules en conflit avec \mathbf{p} (resp. incidentes à \mathbf{p}). Nous exploitons cette propriété de localité pour mettre à jour le résultat de la convection sans relancer le processus de convection à partir de l'enveloppe convexe de l'ensemble de points. La figure 5.2 donne un exemple simple avec un échantillon de points en deux dimensions, dans lequel un seul nouveau point a été ajouté.

Au début du processus de reconstruction sélective initial, la surface s'appuie sur une boîte englobante de P , donc sur une enveloppe convexe. Toutes les cellules de la triangulation de Delaunay \hat{D} sont

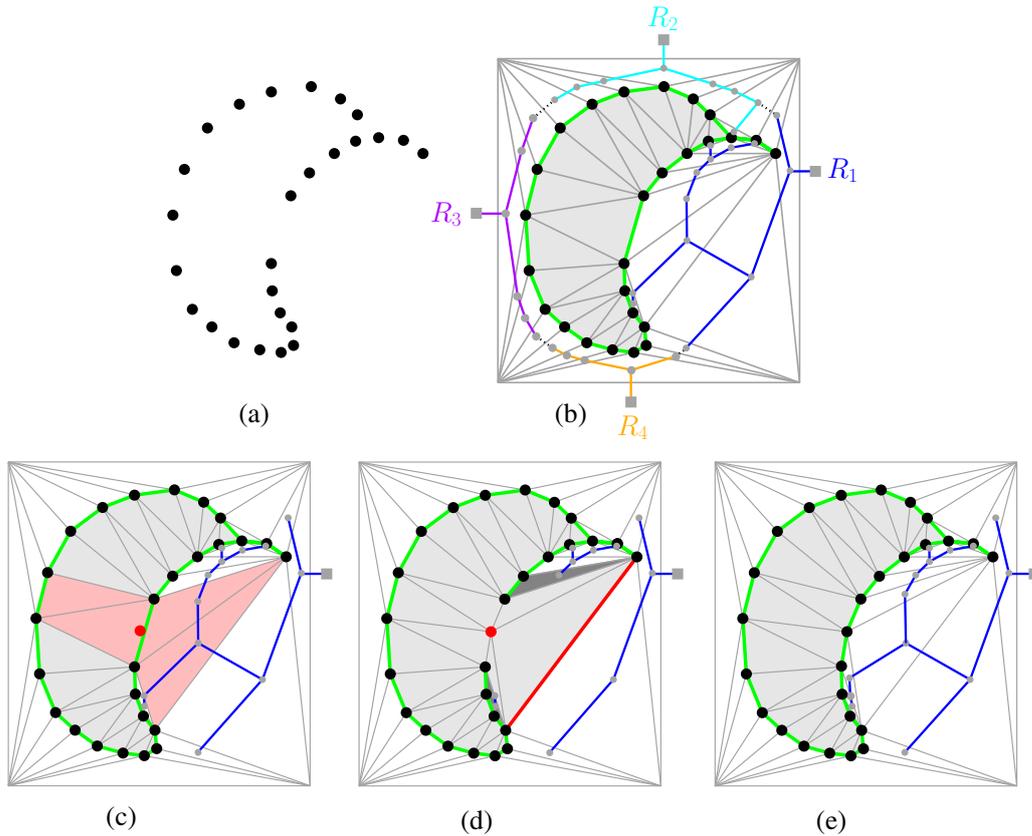


FIG. 5.2 – Mise à jour locale d’une courbe reconstruite par convection géométrique à partir d’un échantillon de points 2D (a). Le résultat de la reconstruction initiale est illustré en (b) avec les quatre relations de découverte sur les cellules de la triangulation de Delaunay traversées. En (c), un nouveau point (rouge) est inséré ; la région de conflit est constituée de l’ensemble des cellules rouges. En (d), la triangulation est mise à jour, brisant la relation R_1 et donnant lieu à des cellules orphelines (grises) ; le processus de reconstruction relancé à partir de l’arête rouge. La courbe mise à jour, avec la relation R_1 restaurée apparaît en (e).

internes, exceptées les cellules infinies, qui sont *externes* à la surface. Rappelons que les cellules infinies sont artificiellement créées pour que les facettes de l’enveloppe convexe soient incidentes à deux cellules. Durant la convection, la surface évolue et les cellules traversées deviennent externes. On dira qu’une cellule externe C_2 a été *découverte* à partir d’une cellule C_1 si C_2 est devenue externe quand la facette incidente à C_1 et C_2 , orientée vers C_2 , a été ouverte par le processus de convection. Une cellule C_2 peut être découverte seulement une fois. Dans le cas où une facette orientée d’une cellule C_3 vers une cellule C_2 est ouverte vers C_2 et que C_2 a déjà été ouverte, cela signifie que la surface a localement deux facettes couplées qui s’effondrent entre C_2 et C_3 . La cellule C_2 est alors dite *redécouverte* à partir de la cellule C_3 .

Le graphe qui représente la relation de découverte entre les cellules de la triangulation de Delaunay est une forêt d’arborescences

$$F = \bigcup_i^m R_i$$

où chaque arborescence R_i est enracinée sur une cellule infinie, dont le nombre m est le nombre de facettes de l'enveloppe convexe de l'ensemble des points insérés dans la surface. Dans le cas d'une initialisation avec une boîte englobante, on a $m = 12$. Ces arborescences constituent une forme d'*historique* de l'évolution de la surface au cours du processus de convection. Une idée clé de l'algorithme de mise à jour locale est de maintenir cet historique dans la triangulation de Delaunay des points conservés. Il est important de noter que cette information n'est pas de nature globale : il s'agit de l'ordre relatif dans lequel la surface a traversé les cellules de la triangulation de Delaunay des points insérés dans la surface.

Une forêt donnée F n'est pas unique dans le mesure où il peut exister plusieurs configurations équivalentes de la relation de découverte selon l'ordre dans lequel les cellules de Delaunay sont explorées. Étant donné une cellule C_2 découverte à partir d'une cellule C_1 et redécouverte à partir d'une cellule C_3 , les relations de découverte et de redécouverte peuvent être échangées si et seulement si C_2 n'est pas un ancêtre de C_3 (figure 5.3).

Lorsque des points sont insérés ou retirés dans la triangulation, des cellules disparaissent. Celles-ci seront appelées cellules *en conflit* dans les deux cas. Elles sont remplacées par de *nouvelles* cellules résultant de la retriangulation locale. Cette retriangulation altère l'intégrité de la relation de découverte. Les cellules externes qui ne disparaissent pas et qui ont été découvertes précédemment à partir d'une cellule en conflit deviennent racines de relations de découverte, bien qu'elles ne soient pas des cellules infinies. Ces cellules sont appelées cellules *orphelines*. Pour relancer le processus de convection, la surface est « regonflée » de manière à ce que les nouvelles cellules créées deviennent internes (figure 5.4). La surface est alors composée des trois types de facettes :

- Les facettes de *relance*, orientées vers les nouvelles cellules internes et qui encodaient précédemment des relations de découverte ou de redécouverte.
- Les facettes *réversibles*, orientées depuis de nouvelles cellules internes vers des cellules orphelines.
- Les autres facettes.

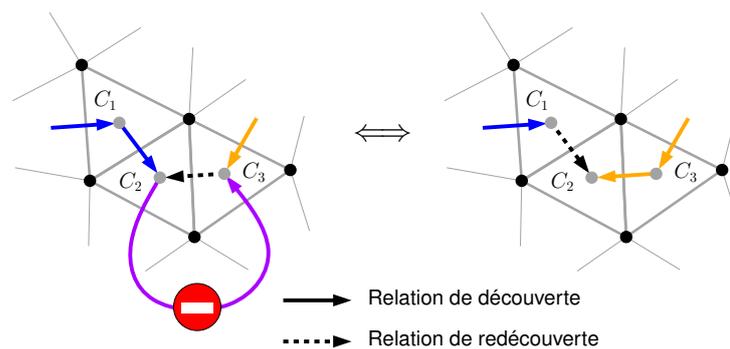


FIG. 5.3 – Equivalence entre deux configurations de cellules en 2D. Les relations de découverte et de redécouverte peuvent être échangées si cela ne crée pas de circuit.

Le processus de mise à jour consiste à restaurer de manière cohérente la relation de découverte au regard de la nouvelle configuration des cellules induite par le nouvel ensemble de points. Ce processus comporte trois étapes :

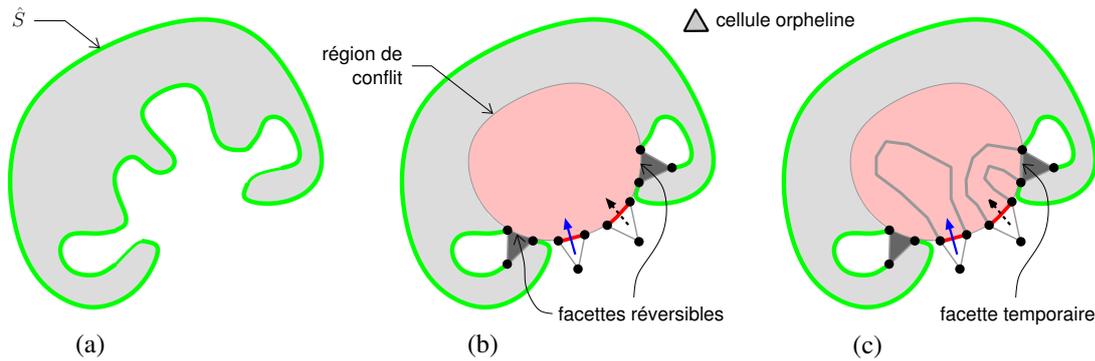


FIG. 5.4 – Convection à travers une région de conflit en 2D. La surface initiale est présentée en (a). En (b), la convection démarre à partir de deux point d’entrée, correspondant respectivement à une relation de découverte et une relation de redécouverte. Les cellules grises sont des cellules orphelines. En (b), la convection à travers la région de conflit s’est arrêtée. Une facette temporaire est apparue.

1. Le processus de convection est d’abord lancé à partir des facettes de relance et restreint à la région de conflit. La surface s’arrête temporairement à la frontière de cette région, en attente de l’opportunité de rétablir une relation de découverte ou de redécouverte vers l’extérieur. Les facettes en attente sont dites *temporaires* (figure 5.4).

2. La connectivité de la relation de découverte doit être rétablie de manière à ce que les cellules orphelines soient découvertes, tout en maintenant une forêt. La création de circuits doit être évitée. Si ce n’est pas possible pour une cellule orpheline donnée, cela signifie que la cellule ne peut être externe dans la nouvelle triangulation de Delaunay, et la surface va reculer localement.

Soit C une cellule orpheline. Supposons qu’il existe une cellule C_1 adjacente à C , à l’extérieur de la surface et telle que C pourrait être découverte depuis C_1 . Si C n’est pas ancêtre de C_1 pour la relation de découverte, une relation de découverte est créée entre C_1 et C . Maintenant que C a été découverte, les facettes qui séparent C de l’intérieur de la surface peuvent être découvertes par le processus de convection. Ces facettes sont ajoutées à l’ensemble des facettes temporaires.

Si une des cellules candidates pour être C_1 est une nouvelle cellule (i.e. à l’intérieur de la région retriangulée), alors elle est choisie plutôt que les autres. S’il n’y a pas de cellule candidate pour découvrir C , alors la cellule C devient interne et la surface recule. Si une cellule C_2 a été découverte à partir de C , alors elle devient une cellule orpheline (si elle n’est pas infinie), qui sera traitée à son tour. Ce processus conduit à la restauration de la relation de découverte.

3. La dernière étape consiste à lancer le processus de convection à partir des facettes temporaires bloquées artificiellement pendant la restauration de la relation de découverte.

Une caractéristique importante de l’algorithme de mise à jour locale est que le résultat ne dépend ni de conditions d’échantillonnage, ni de l’ordre d’insertion ou de retrait de plusieurs points.

La mise à jour locale dynamique peut avoir de nombreuses applications. Elle peut par exemple être utilisée pour améliorer la qualité de l’échantillonnage et de la forme des triangles d’une surface reconstruite. Nous développons ici une application de correction interactive.

5.2 Application à la correction interactive

Selon la tolérance d'erreur choisie, les modèles reconstruits de façon automatique à l'issue d'une phase de reconstruction sélective peuvent être trop grossiers localement, ou au contraire comporter certains détails non désirés pour certaines applications. Il peut donc être intéressant de pouvoir corriger le niveau de précision de ces modèles a posteriori de manière locale sans repartir à zéro. En outre, la visualisation interactive de modèles de grande taille peut nécessiter d'adapter le niveau de précision de ceux-ci au point de vue dans une scène pour faire face à des limites matérielles.

5.2.1 Algorithme principal

En nous appuyant sur les algorithmes de reconstruction sélective et de mise à jour locale, nous proposons une technique de correction interactive consistant à modifier localement la valeur du paramètre ρ_{geom} et à mettre à jour la surface reconstruite en fonction des nouvelles conditions d'échantillonnage locales. En étant plus ou moins restrictif localement sur les variations de la normale, il est possible d'ajouter des données ou d'en éliminer avec un contrôle intuitif.

Notre algorithme de correction interactive prend en entrée une surface reconstruite S_{simp} dont le niveau de précision est à modifier dans une région incluse dans un volume englobant \mathcal{R} . Nous entendons par ρ_{geom} la valeur du paramètre de simplification associée à la reconstruction de S_{simp} et $\rho_{geom}^{\mathcal{R}}$ la valeur de ce paramètre associée à la région de S_{simp} à mettre à jour. L'algorithme procède comme suit :

1. Les points de l'échantillon initial P se trouvant dans la région \mathcal{R} qui ont été éliminés lors de la phase de reconstruction sélective précédente sont réhabilités.
2. Les cellules de la triangulation de Delaunay de l'échantillon de points P_{simp} associé à S_{simp} dont la sphère circonscrite intersecte \mathcal{R} sont détectées, formant une région de conflit \mathcal{C} telle que $\mathcal{R} \subseteq \mathcal{C}$.
3. Le processus de reconstruction sélective est relancé à partir de la frontière de \mathcal{C} selon l'algorithme de mise à jour locale, en appliquant la procédure de simplification avec la valeur $\rho_{geom}^{\mathcal{R}}$ comme paramètre de simplification.

L'algorithme est illustré par la figure 5.5. En pratique, \mathcal{R} peut être défini simplement comme une boîte ou une sphère. La région de conflit \mathcal{C} peut être construite efficacement en localisant une cellule de Delaunay contenant un point de \mathcal{R} , par exemple son centre, puis en testant récursivement les intersections entre \mathcal{R} et les sphères circonscrites des cellules de Delaunay voisines de la première. Pour éviter une discontinuité dans l'échantillonnage, la procédure de simplification est appelée avant l'étape 3 pour tous les sommets situés sur la frontière de \mathcal{C} avec la valeur $\rho_{geom}^{\mathcal{R}}$ comme paramètre de simplification. Précisons enfin que Les sommets de la triangulation de Delaunay internes à \mathcal{C} sont conservés, et servent de balises pour accélérer la reconstruction locale.

Un inconvénient de la méthode est que le niveau de précision change de manière abrupte entre les régions reconstruites avec différents niveaux de précision, ce qui nuit à la qualité du maillage (fig. 5.6(b)). Pour obtenir une transition progressive dans la densité de l'échantillon (fig. 5.6(c)), nous spécifions le

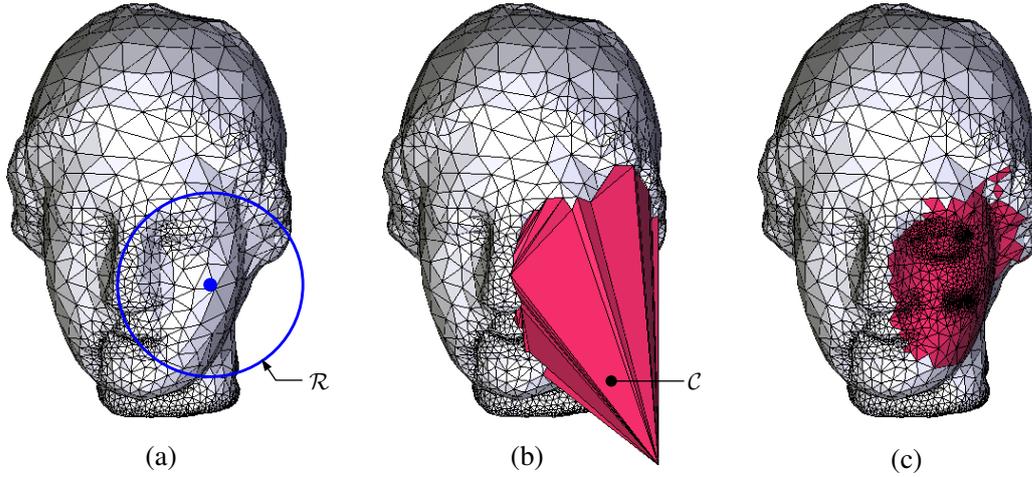


FIG. 5.5 – Illustration de l'algorithme de correction interactive sur le modèle IGEA. En (a), la région de mise à jour est définie. L'image en (b) montre le modèle avec la frontière externe de la région de conflit correspondante. En (c) apparaît le résultat de la correction, avec les facettes résultant de la mise à jour locale mises en évidence.

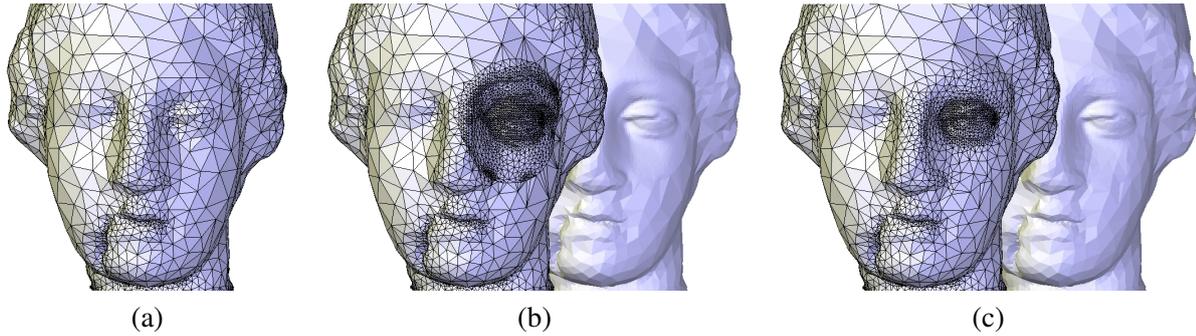


FIG. 5.6 – Raffinement interactif du modèle IGEA. En (a), le modèle a été reconstruit avec $\rho_{geom} = 0.75$. En (b), la surface a été raffinée dans une région sphérique autour de l'oeil gauche avec une valeur constante $\rho_{geom}^{\mathcal{R}} = 0.95$ dans toute la région. En (c), la même région a été raffinée en utilisant la fonction $\delta_{geom}^{\mathcal{R}}$ autour du centre de la région sphérique.

niveau de précision en tout point de la région de mise à jour \mathcal{R} en interpolant entre ρ_{geom} et $\rho_{geom}^{\mathcal{R}}$ au moyen d'une fonction potentiel de simplification définie à partir d'un squelette. Formellement, nous définissons cette fonction $\delta_{geom}^{\mathcal{R}} : \mathbb{R}^3 \rightarrow [0, 1]$ par :

$$\delta_{geom}^{\mathcal{R}}(\mathbf{x}) = g \circ d(\mathbf{x})$$

où $d : \mathbb{R}^3 \rightarrow \mathbb{R}_+$ est une fonction de distance à un squelette et $g : \mathbb{R}_+ \rightarrow [0, 1]$ est une fonction de distribution du potentiel dans l'espace que nous définissons comme suit :

$$g(r) = \rho_{geom} + \frac{\rho_{geom}^{\mathcal{R}} - \rho_{geom}}{R^{2n}} (R^n - r^n)^2$$

où $R \geq 0$ est un rayon d'influence et $n \geq 2$ une constante de raideur. La fonction g varie de façon

monotone avec $g(0) = \rho_{geom}^{\mathcal{R}}$ en tout point du squelette et $g(r) = \rho_{geom}$ pour tout $r \geq R$. La fonction g est C^2 et admet une demi-tangente horizontale en 0 et en R , ce qui est destiné à assurer un gradient de densité continu entre la région mise à jour et le reste de la surface reconstruite. Lorsque le processus de reconstruction sélective est relancé localement dans la région de conflit associée à \mathcal{R} , la procédure de simplification pour tout point $\mathbf{p} \in P \cap \mathcal{R}$ incorporé dans la surface en convection est désormais $\delta_{geom}^{\mathcal{R}}(\mathbf{p})$. Si $\rho_{geom}^{\mathcal{R}} > \rho_{geom}$, alors l'échantillon sera raffiné ; sinon, la surface obtenue sera localement plus grossière. La figure 5.7 illustre le comportement de la reconstruction pour différentes valeurs des paramètres de simplification et du paramètre de raideur de la fonction $\delta_{geom}^{\mathcal{R}}$. Le squelette utilisé pour l'évaluation de la fonction de distance est un point correspondant au centre d'une région sphérique.

5.2.2 Corrections locales multiples

Avec la fonction potentiel de simplification précédente, la surface ne peut être mise à jour dans plusieurs régions \mathcal{R} sans créer de discontinuité dans l'échantillonnage que si les régions \mathcal{R} sont disjointes. Pour fusionner différentes régions de mise à jour qui se recouvrent avec une transition lisse dans la densité d'échantillonnage, notre algorithme procède en sommant les fonctions potentiel de simplification associées à ces régions. Un champ de potentiel global nécessite alors d'être défini sur l'ensemble des points de l'échantillon initial P ; nous noterons $v(\mathbf{p})$ la valeur de potentiel de simplification associée à un point $\mathbf{p} \in P$. Avant de lancer le processus de reconstruction sélective initial, la valeur ρ_{geom} choisie est d'abord affectée à tous les points de P . Ensuite, lorsque la reconstruction est mise à jour dans une région \mathcal{R} , le potentiel calculé pour chaque point $\mathbf{p} \in P \cap \mathcal{R}$ est

$$v(\mathbf{p}) \leftarrow \max\{v(\mathbf{p}) + \delta_{geom}^{\mathcal{R}}(\mathbf{p}), 1\}.$$

La figure 5.8 montre l'application de ce principe au modèle Igea, sur lequel nous avons effectué plusieurs opérations locales de raffinement.

5.2.3 Interface de correction interactive

Nous avons développé une interface graphique qui permet de charger un modèle reconstruit de manière sélective à partir d'un fichier décrivant le maillage et de modifier interactivement son niveau de précision localement en utilisant notre outil de correction (figure 5.12). L'utilisateur peut ensuite spécifier les zones à raffiner ou à rendre plus grossières à l'aide de sphères de sélection. La valeur locale du paramètre de simplification est déterminée par l'utilisateur en connaissant le niveau de précision du modèle initial, indiqué dans le fichier contenant le maillage. Le maillage résultant d'une mise à jour peut ensuite être stocké pour être éventuellement à nouveau modifié plus tard.

Détails d'implantation Le résultat d'une reconstruction sélective est stocké sous la forme d'un maillage surfacique, dans une représentation indexée où les sommets ne sont pas représentés par leur

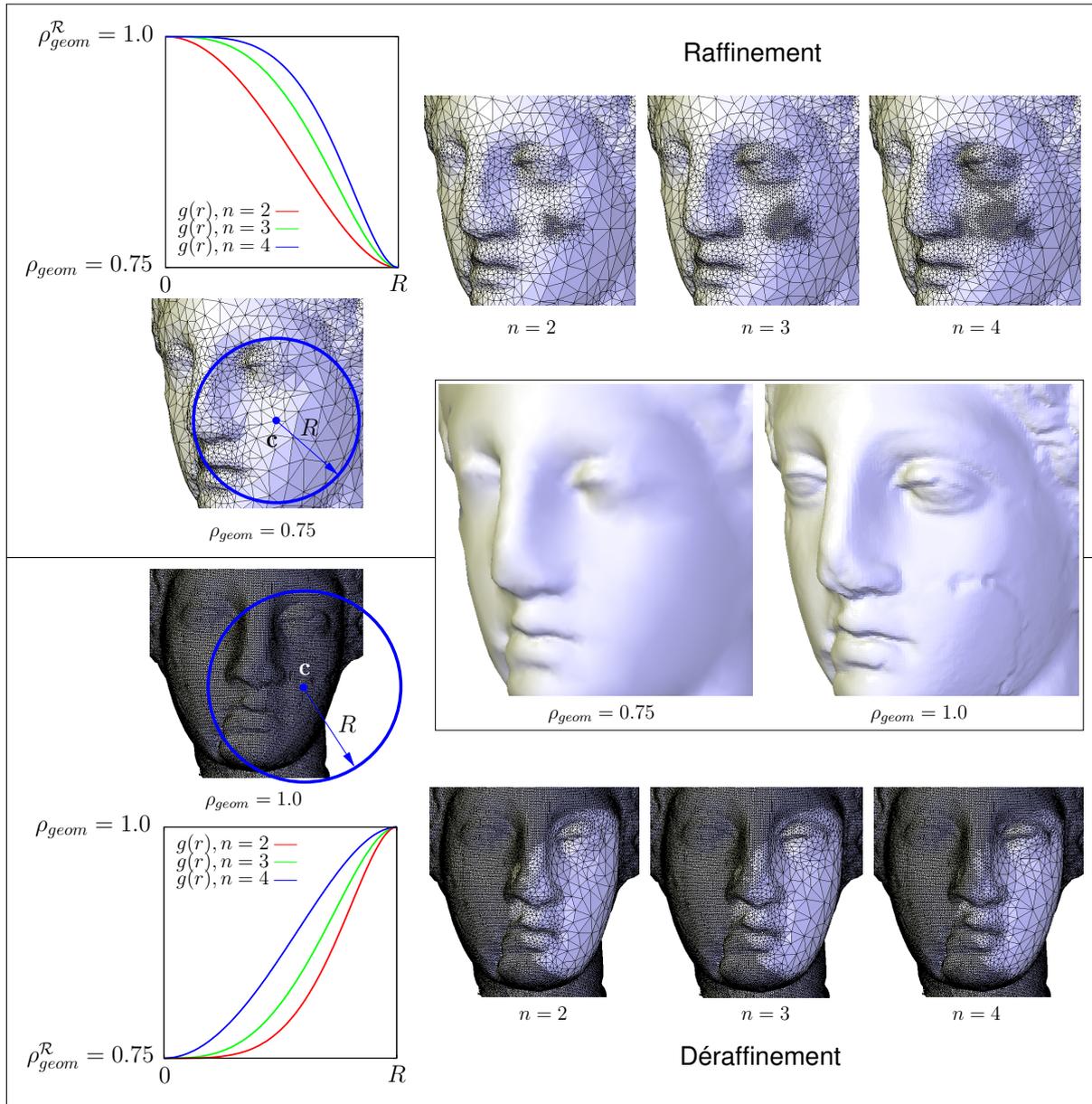


FIG. 5.7 – Correction interactive sur le modèle IGEA. Des résultats d'opérations de raffinement (en haut) et de déraffinement (en bas) obtenus à l'aide de fonctions $\delta_{geom}^{\mathcal{R}}$ avec différentes valeurs de raideur sont présentés.

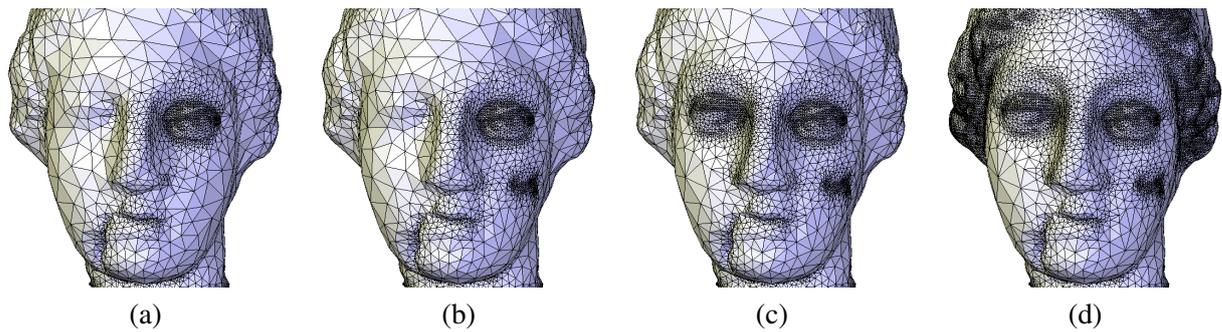


FIG. 5.8 – Fusion de plusieurs régions de mise à jour locale sur le modèle IGEA. La reconstruction initiale est celle de la figure 5.6. Un premier raffinement a été effectué autour de l’œil gauche en (a), puis un second sur la joue gauche en (b), puis troisième en (c) sur l’œil droit, en enfin huit régions différentes ont été fusionnées avec différentes valeurs de ρ_{geom}^R pour obtenir le résultat en (d).

géométrie mais par l’indice des points correspondants dans le fichier contenant l’échantillon de points original. Au cours de la lecture du fichier décrivant le maillage, le fichier contenant les points est lu en parallèle, et la triangulation de Delaunay des points associés aux sommets est construite. Un kd -arbre stockant tous les points de l’échantillon est également construit. Ensuite, le processus de reconstruction par convection géométrique classique est lancé à travers cette triangulation en testant si les facettes traversées sont terminales ou non relativement à la description du maillage. Chaque sommet de la triangulation stocke l’indice du point qui lui est associé, et les tests d’ouverture de facettes sont accomplis en utilisant une table de hachage dont les clés sont les triplets d’indices des facettes du maillage. Au cours de ce processus, qui constitue un processus de reconstruction simplifié, des relations de découverte sont établies entre les cellules de la triangulation de Delaunay traversées, sur lesquelles s’appuient ensuite les mises à jour locales. Avec un modèle de base relativement grossier, ce processus d’initialisation est très rapide, de l’ordre de quelques secondes pour des modèles comportant jusqu’à 100k sommets. Lorsqu’un modèle est considéré comme définitif, un maillage indépendant de l’échantillon de points original peut facilement être généré.

5.3 Résultats expérimentaux

Nous avons implanté notre outils de correction interactive en proposant deux types de squelette : le point et le segment, avec la distance euclidienne comme fonction de distance au squelette. Les figures 5.8, 5.9, 5.10 et 5.11 illustrent l’application de cet outil sur divers modèles. Les temps de calcul sont répertoriés dans la table 5.1.

L’utilisateur peut complètement personnaliser la reconstruction pour mettre en valeur certains détails. Pour quelques dizaines de milliers de points, la mise à jour peut s’effectuer en temps interactif (inférieur à 10 secondes). Cependant, le coût d’une mise à jour dépend en partie de la complexité de la région de conflit, et plus particulièrement de la hauteur de l’arborescence de la relation de découverte à restaurer.

L’implantation actuelle pourrait évoluer pour éviter de stocker tous les points en mémoire. Seuls les

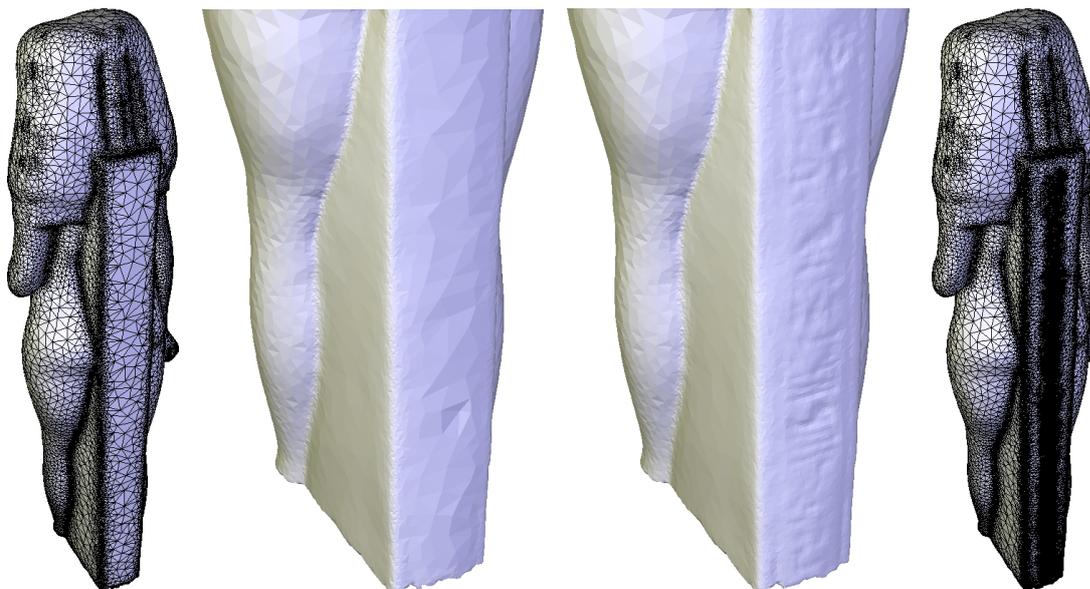


FIG. 5.9 – Correction interactive sur le modèle ISIS. Le modèle original possède des hiéroglyphes gravés au dos, qui ne sont pas capturés pour une valeur de ρ_{geom} trop petite. Un raffinement local permet de les faire apparaître.

points des régions à mettre à jours pourraient être chargés en mémoire. C'est une voie que nous avons dans un premier temps exploré pour une reconstruction sélective initiale.

Modèle		Reconstruction initiale			Mise à jour locale	
Nom	$ P $	ρ_{geom}	$ P_{simp} $	Temps	$ P'_{simp} $	Temps
TRIPLE HÉCATE	90 180	0.85	30 832	0 :22	34 263	0 :01
ISIS	187 644	0.9	22 368	0 :36	33 892	0 :16
ASIAN DRAGON	3 609 600	0.65	185 504	2 :48	177,324	0 :14
DAVID	3 617 008	0.6	137 025	2 :06	174 628	0 :28

TAB. 5.1 – Performances de notre méthode de correction interactive pour différents échantillons de points. Les temps sont données en minutes :secondes. Tous les tests ont été effectués sur un processeur Pentium IV 3.0 GHz avec 2 Go de RAM.

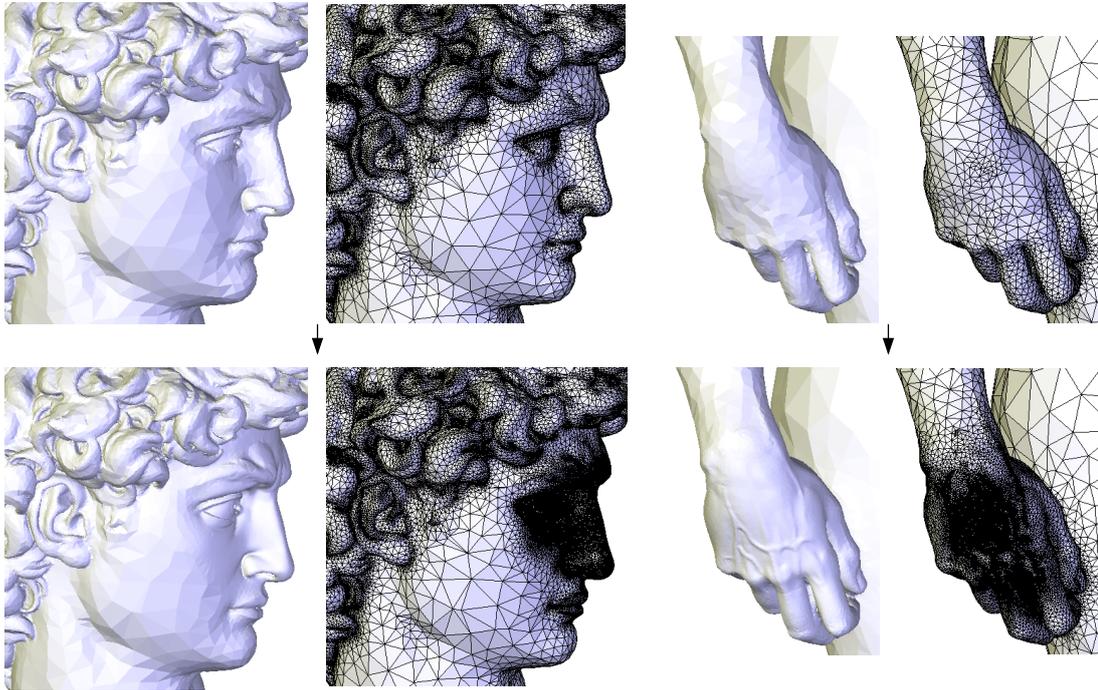


FIG. 5.10 – Correction interactive sur le modèle DAVID. Les régions de l’œil droit et de la main droite ont été raffinées pour faire apparaître les détails de l’œil et des veines.

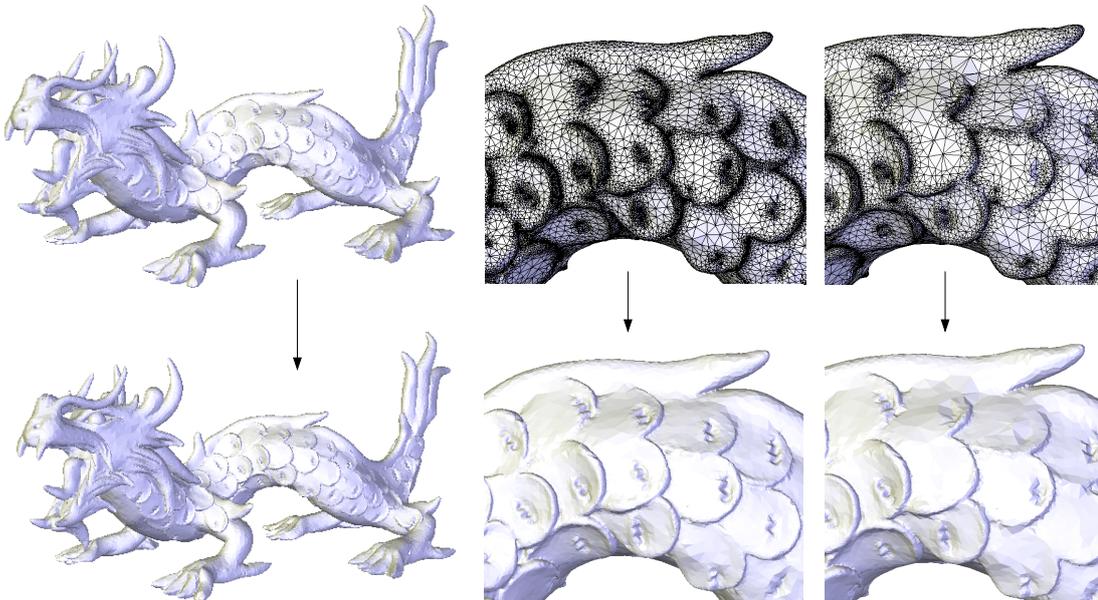


FIG. 5.11 – Correction interactive sur le modèle DRAGON. Le niveau de précision a été réduit localement au niveau des écailles sur le dos.

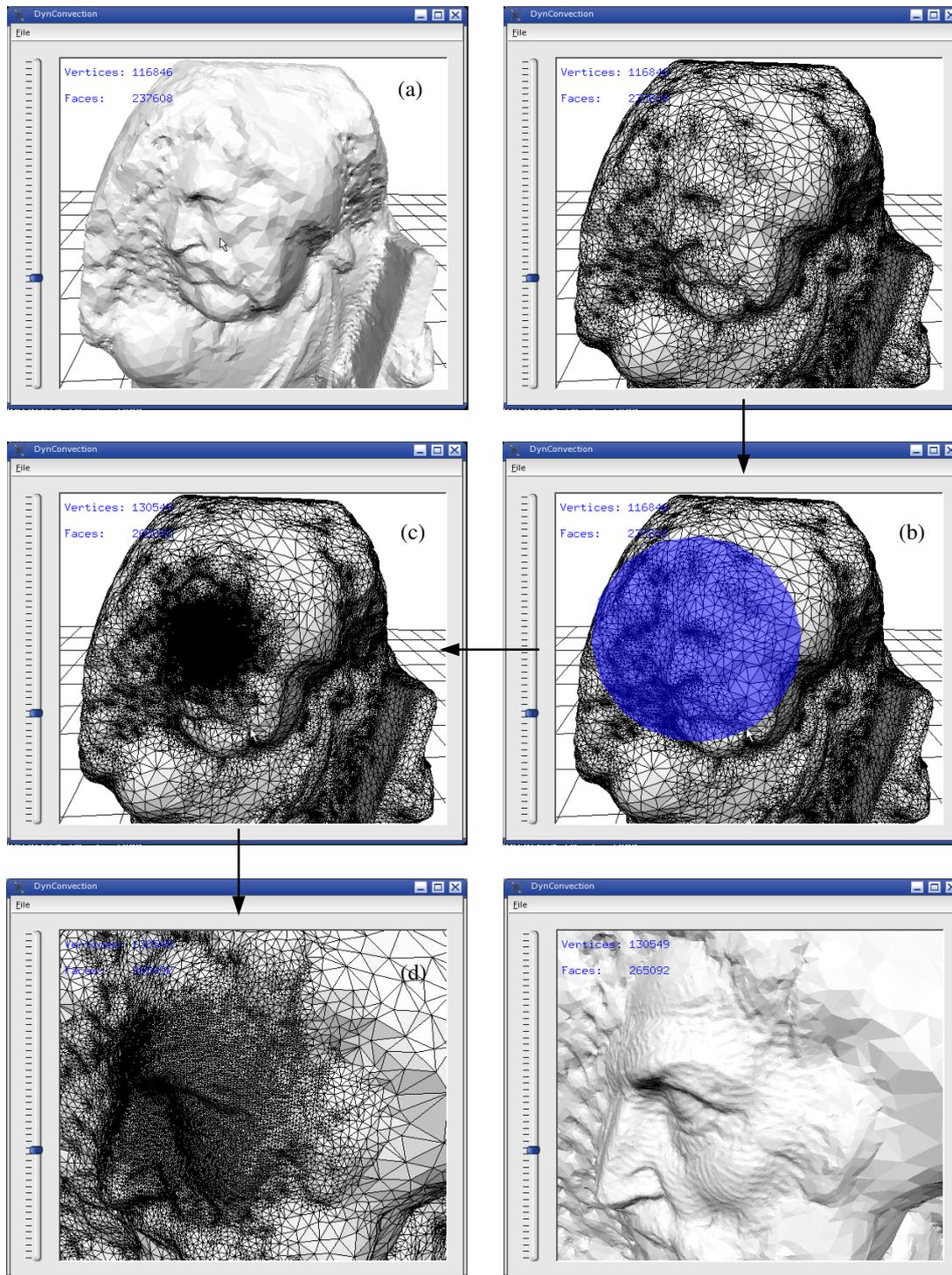


FIG. 5.12 – Captures de notre interface de correction interactive. En (a), un modèle initial obtenu par reconstruction sélective a été chargé. En (b), une région à raffiner a été définie. Le résultat de la mise à jour apparaît en (c), avec des zooms sur la zone raffinée en (d).

Reconstruction sélective par subdivision-fusion

Les algorithmes que nous avons proposés jusqu'ici considèrent que toutes les données d'entrée sont présentes en mémoire centrale. Notre dernier objectif est de reconstruire des surfaces simplifiées à partir de données qui ne tiennent pas entièrement en mémoire.

Beaucoup d'algorithmes géométriques pour des données non entièrement en RAM procèdent selon le principe bien connu de division et fusion. Les données d'entrée sont partitionnées en blocs pouvant être traités indépendamment en mémoire centrale, puis les résultats locaux sont fusionnés pour obtenir le résultat global. Cette stratégie a été utilisée en particulier pour simplifier de gros maillages non structurés [129, 60, 59]. Cependant, celle-ci ne s'étend pas facilement à la reconstruction de surface à partir d'échantillons de points non structurés. L'absence d'information de connectivité entre les différents blocs entraîne des problèmes de raccordement et d'orientation [78].

Dans ce chapitre, nous proposons une extension de notre méthode de reconstruction sélective selon le principe de division et fusion pour traiter des échantillons de points ne tenant pas entièrement en mémoire centrale. L'idée clé est le maintien d'une triangulation de Delaunay unique. Notre solution consiste à partir d'une reconstruction initiale grossière, pour ensuite la raffiner localement à l'aide de l'algorithme de reconstruction sélective par mise à jour locale, en considérant une décomposition des données en blocs. Une limite de la méthode actuelle est que le résultat final doit tenir en mémoire.

6.1 Description de l'algorithme

Étant donné un échantillon de points P ne tenant pas entièrement en mémoire centrale, l'idée est de charger un à un en RAM des blocs de points situés dans une même région de l'espace en étendant la reconstruction au fur et à mesure. Si on considère une telle approche en laissant de côté la question de la simplification, on constate que le résultat ne peut pas être obtenu par une unique passe sur chacun des blocs du fait d'une dépendance vis à vis de l'ordre de chargement des données (fig. 6.1). Notre algorithme procède selon les trois étapes suivantes, schématisées dans la figure 6.2.

1. L'échantillon de points en entrée P est filtré à travers une grille pour obtenir un sous-ensemble de points représentatifs P_{rep} et une partition de P en blocs de points $P_1 \cup P_2 \cup \dots \cup P_n = P$.

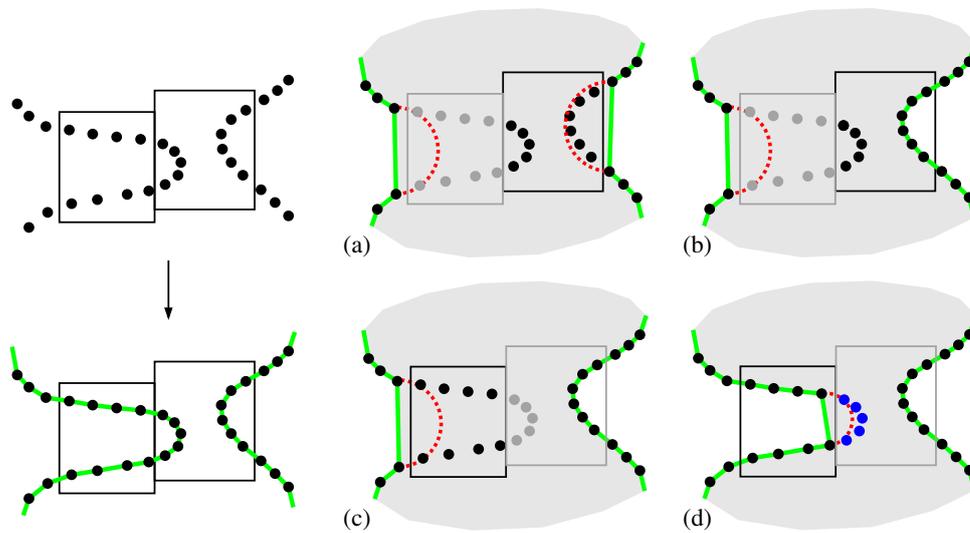


FIG. 6.1 – Illustration de la problématique de dépendance du processus de convection géométrique à l’ordre de chargement de blocs de données. En (a), les points contenus dans le bloc de droite sont chargés en premier, permettant à la surface en convection d’évoluer pour atteindre l’état en (b). En (c), les données du bloc de gauche sont chargées, la surface atteignant alors l’état en (d). Plusieurs points du bloc de droite (en bleu) n’ont pas été atteints par la surface. En chargeant le bloc de gauche en premier, tous les points auraient été atteints.

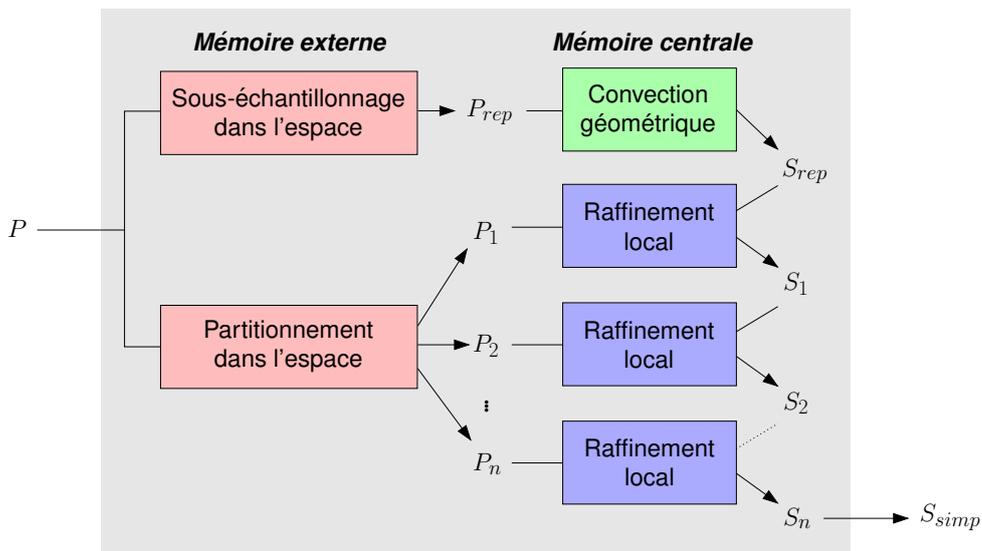


FIG. 6.2 – Processus de reconstruction sélective selon une approche division et fusion pour traiter des données ne tenant pas entièrement en mémoire centrale.

2. La triangulation de Delaunay de P_{rep} est construite et l'algorithme de convection géométrique classique est appliqué à cet échantillon de points pour produire une reconstruction initiale S_{rep} .
3. Pour chaque sous-ensemble de points P_i , les points qu'il contient sont chargés en mémoire centrale, puis la surface S_{rep} est raffinée localement dans la boîte englobante de P_i à l'aide de l'algorithme de reconstruction sélective, par mise à jour locale.

Les deux premières étapes de la méthode sont illustrées par la figure 6.3. En partitionnant l'échantillon de points en entrée dans la première étape, notre objectif est d'obtenir rapidement un sous-ensemble de points P_{rep} donnant une idée approximative, même inexacte, de la forme globale de l'objet. Cet échantillon est ensuite utilisé dans la deuxième étape pour produire la reconstruction grossière S_{rep} . Nous n'avons ici pas besoin d'un sous-échantillonnage précis. L'intérêt de cette étape est en effet surtout pour les relations de découverte établies entre des cellules d'une triangulation de Delaunay qui partitionne le domaine des données. Ces relations vont servir de base aux étapes de mise à jour locale de la reconstruction. Même si la reconstruction initiale n'est pas topologiquement correcte et ne capture pas certains détails, ceci n'affectera pas la qualité du résultat final ; les erreurs seront automatiquement réparées par les mises à jour. Il est cependant préférable de disposer d'un échantillon suffisamment dense pour des raisons d'*accessibilité*. Dans la reconstruction initiale S_{rep} , il est possible que certaines concavités de la surface échantillonnée sous-représentées dans P_{rep} ne soient pas atteintes par la surface en convection, ce qui peut bloquer localement le processus de mise à jour lors du chargement des blocs et nécessiter des chargements multiples pour certains. Une condition suffisante pour n'avoir à charger chaque bloc qu'une seule fois est que la surface S_{rep} soit telle que tous les points de l'échantillon P soient compris

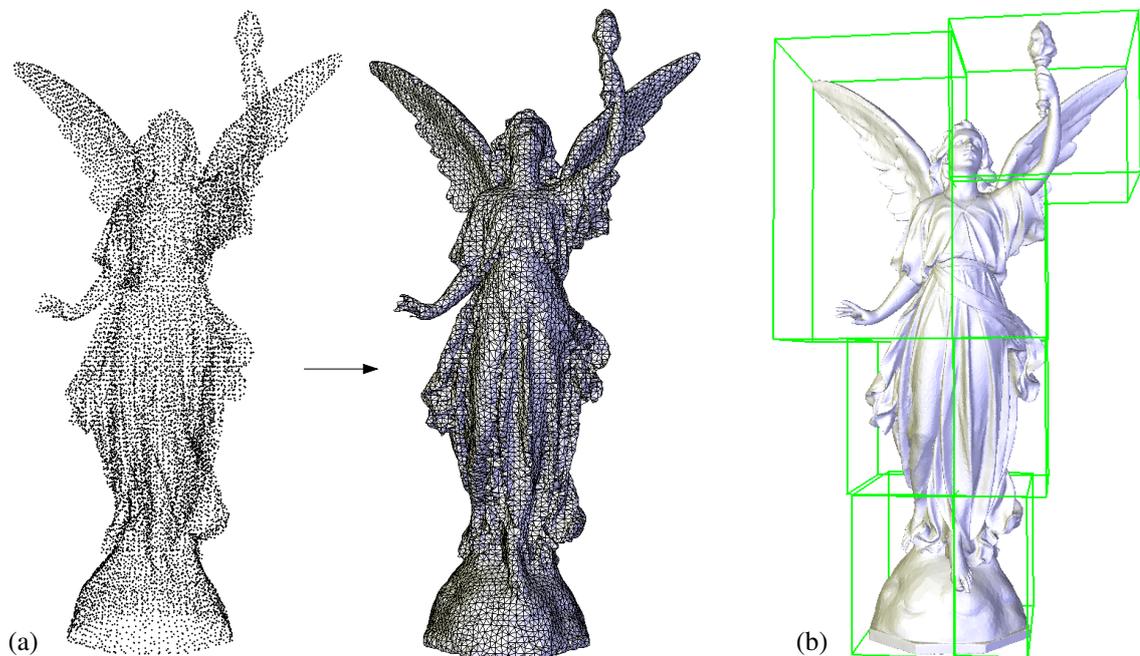


FIG. 6.3 – Illustration des étapes 1 et 2 sur le modèle LUCY (14M points). En (a) apparaît le résultat d'une phase de filtrage et la reconstruction initiale associée. Ici, le sous-échantillon de représentants P_{rep} comporte environ 25k points, et la décomposition comporte 7 blocs.

dans l'union des demi-sphères de Gabriel des facettes de la surface en convection. Cette condition n'est pas facilement vérifiable en pratique, mais le filtrage de P sur une grille de résolution fixée à quelques pourcents de la taille de la boîte englobante suffit à donner de bons résultats. Il pourrait être intéressant de déterminer un critère donnant une résolution optimale.

La phase initiale de filtrage et de partitionnement est accomplie en lisant l'échantillon de points trois fois. Au cours de la première passe de lecture, nous calculons la plus petite boîte englobante, qui est ensuite subdivisée en une grille régulière. Dans la deuxième passe, nous calculons pour chaque cellule non vide de la grille le point de l'échantillon le plus proche du centre. Cet ensemble de points forme l'ensemble P_{rep} . Au cours de cette passe, nous comptons aussi combien de points se trouvent dans chaque cellule de la grille. Nous définissons ensuite une partition binaire de la structure de grille avec un nombre maximum de points dans chaque feuille spécifié par l'utilisateur ; chacune de ces feuilles représente un bloc de points P_i . Le seuil de population maximum pour chaque bloc devrait être établi en fonction de la quantité de mémoire disponible sur la machine sur laquelle s'exécute l'algorithme de reconstruction. Durant une troisième passe, les points sont distribués dans les différentes cellules feuilles. Les ensembles de points contenus dans les différents blocs peuvent être déterminés puis écrits dans des fichiers indépendants sur disque au cours d'une unique lecture du fichier d'entrée.

Dans l'étape 3, nous devons déterminer pour chaque bloc P_i l'ensemble des cellules de la triangulation de Delaunay courante qui entrent en conflit avec les points qu'elle contient. Pour éviter des localisations multiples dans la triangulation de Delaunay, les conflits sont testés à partir de la boîte englobante de P_i . Nous cherchons les cellules de Delaunay dont la sphère circonscrite intersecte cette boîte englobante. Celles-ci sont obtenues en localisant d'abord la cellule de Delaunay qui contient le centre de la boîte, puis en faisant croître la région de conflit en testant récursivement les cellules voisines. Le résultat est un ensemble connexe de cellules de Delaunay utilisé pour initialiser le processus de mise à jour locale de la reconstruction. Pour accomplir les requêtes spatiales efficacement, un k d-arbre des points contenus dans la région de conflit est construit. Cet ensemble inclut P_i et peut aussi contenir des points à l'extérieur de P_i correspondant à des sommets de cellules de Delaunay en conflit avec la boîte englobante de P_i , ce qui garantit que les différentes parties raffinées fusionnent correctement. Le processus de mise à jour locale est alors accompli comme décrit précédemment.

Deux étapes du processus de reconstruction sont illustrées sur le modèle LUCY dans la figure 6.4. Les bords des différents blocs peuvent être légèrement perceptibles dans le résultat final. Cependant, la méthode n'induit pas de discontinuité dans la densité de l'échantillonnage. Ces bords pourraient être complètement éliminés en élargissant simplement les blocs pour que ces derniers contiennent des points voisins jusqu'à une distance dépendant du paramètre de simplification ρ_{geom} .

6.2 Résultats expérimentaux

Nous avons testé notre méthode sur deux modèles : LUCY, comportant environ 14M points, et ST. MATTHEW, comportant environ 26M points. Les deux modèles ont été partitionnés de manière à ce que

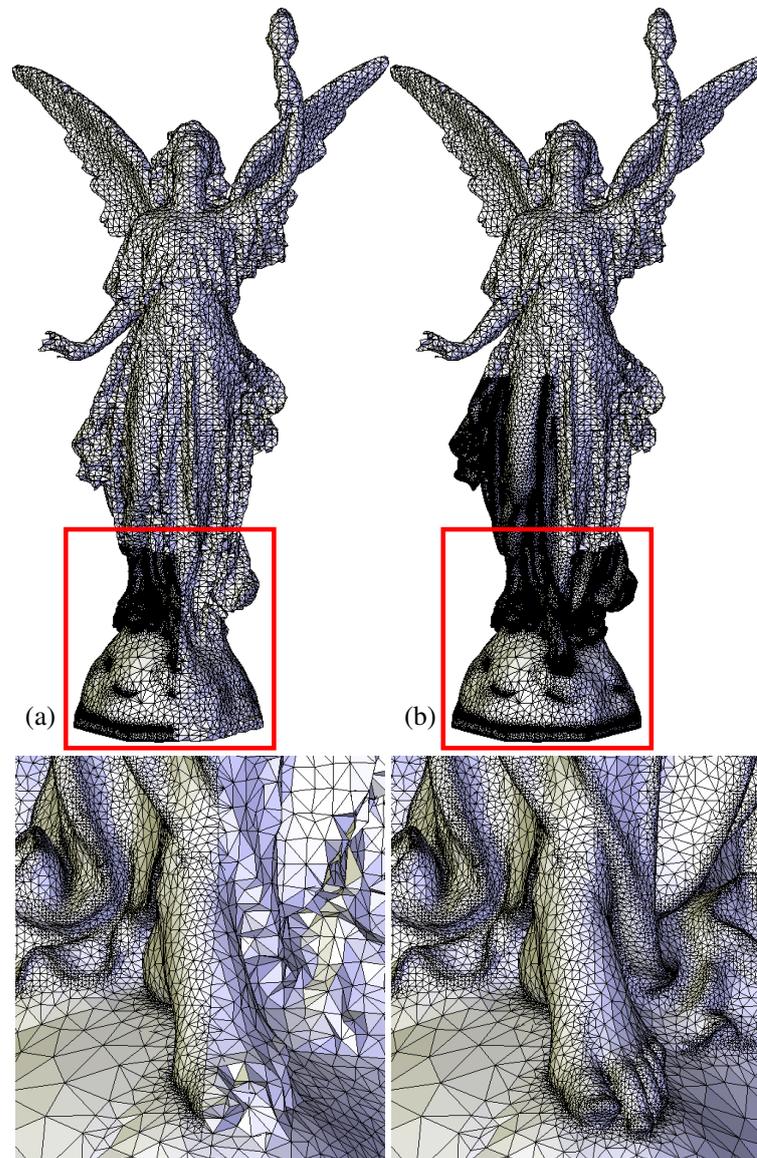


FIG. 6.4 – Deux étapes de reconstruction pour le modèle LUCY. En (a), la surface initiale a été reconstruite à partir des représentants (25k) et une première étape de mise à jour locale a été effectuée. En (b), deux blocs de plus ont été chargés et la reconstruction a été mise à jour.

chaque bloc contienne moins de 3.5M points ; le nombre de blocs était respectivement de 7 et 15. Le nombre initial de points représentatifs a été fixé à 1 pour 1k points et la reconstruction initiale a pris moins de 10 secondes dans les deux cas. Pour chaque bloc, le raffinement de la reconstruction a ensuite pris moins de 2.5 minutes.

Le coût en mémoire comprend le coût du stockage de l'échantillon P_{rep} , puis, pour chaque bloc, le coût du stockage de l'ensemble de points P_i et la structure de kd-arbre associée. Au fur et à mesure des opérations de raffinement, le nombre de points impliqués dans le résultat final augmente, ce qui fait que le coût maximal associé au stockage de la géométrie en mémoire est supérieur à celui de chaque bloc. Enfin, la triangulation de Delaunay construite à mesure constitue le dernier élément à prendre en compte dans le coût de stockage.

Modèle		Reconstruction sélective			
Nom	$ P $	ρ_{geom}	$ P_{simp} $	Temps	Mémoire
LUCY	14 027 872	0.85	550 877	17 :40	765 Mo
ST. MATTHEW	26 034 562	0.5	116 846	31 :12	836 Mo

TAB. 6.1 – Temps de calcul et coût en mémoire pour deux reconstructions sélectives pour des données ne tenant pas entièrement en mémoire. Les temps sont donnés en minutes :secondes et incluent les temps de prétraitement. La colonne « mémoire » correspond à la quantité maximale de mémoire utilisée durant les reconstructions, en mégaoctets. Les tests ont été effectués sur un processeur Pentium IV 3.0 GHz avec 2 Go de RAM.



FIG. 6.5 – Reconstruction sélective des modèles LUCY (14M points), en (a), et ST. MATTHEW (26M points), en (b).

Conclusions et perspectives

Dans cette partie, nous avons proposé un cadre de travail combinatoire dynamique pour la reconstruction de surface à partir d'échantillons de points non structurés. Nous avons tout d'abord mis en place un algorithme de reconstruction sélective permettant de reconstruire directement des surfaces simplifiées à partir d'échantillons de points denses. Dans ce but, nous avons développé une procédure de sous-échantillonnage local spécifique, combinée à un algorithme de reconstruction de surface par convection géométrique. La procédure de simplification repose sur une mesure d'épaisseur locale, dont le sens a été établi dans le cadre continu et dans le cadre discret sous certaines conditions d'échantillonnage. Nous avons en outre étudié les structures de données nécessaires à une mise en œuvre efficace de cette méthode pour des volumes de données importants. Nous avons ensuite présenté un algorithme permettant une mise à jour locale dynamique d'une surface reconstruite en ajoutant ou en retirant des données. Cette dernière fonctionnalité nous a permis de développer des outils de correction interactive de modèles, ainsi qu'une extension de notre algorithme de reconstruction sélective à des échantillons de points de grande taille ne pouvant pas être entièrement chargés en mémoire centrale.

Les différents résultats obtenus sont satisfaisants et traduisent la capacité de notre méthode à apporter de la flexibilité à la reconstruction de surface. Elle présente ainsi de nombreux avantages susceptibles de rendre plus efficace la chaîne de production de modèles 3D.

- Avec des données géométriques moins redondantes, les maillages générés sont plus compacts et de meilleure qualité, et les traitements ultérieurs peuvent être accélérés de manière significative.
- Le résultat d'une reconstruction peut être facilement mis à jour si de nouvelles données sont disponibles ou si des corrections doivent être effectuées.
- L'algorithme de mise à jour locale ne nécessite pas de partir d'un ensemble de points respectant des conditions d'échantillonnage spécifiques et ne dépend pas de l'ordre d'insertion ou de retrait des données (sans simplification), ce qui permet de reconstruire une surface progressivement à partir d'un flux de données, par exemple transmis sur un réseau.

Notre méthode actuelle présente cependant un certain nombre de limites qui touchent à ses performances, à la taille des données qu'elle peut traiter et à sa capacité à fournir de bons résultats de simplification à partir de données bruitées.

- Les performances de l'algorithme de reconstruction sélective sont pénalisées par le nombre de requêtes spatiales liées à l'évaluation du prédicat P_{go} et de la requête Q_{dt} , qui est proportionnel au nombre de cellules de la triangulation traversées par la surface en convection.

- L'extension de l'algorithme à des échantillons de points ne tenant pas entièrement en mémoire vive est limitée dans la mesure où le résultat de la reconstruction sélective doit tenir en mémoire. Cela ne nous permet de traiter des données que sous l'hypothèse d'une certaine redondance, difficile à vérifier a priori. Le maintien d'une triangulation de Delaunay unique au cours du processus est le principal facteur limitant, mais il s'agit du prix à payer pour éviter des opérations de couture et d'orientation consistante.
- En présence de données trop bruitées, l'estimation des normales peut manquer de fiabilité, ce qui retentit directement sur celle de la mesure d'épaisseur locale. En conséquence, les importantes variations de la géométrie locale détectées conduisent à mesurer une faible redondance, ce qui a pour effet une simplification réduite. Nous n'avons appliqué notre méthode à des données bruitées qu'en lissant suffisamment le champ des normales estimées.

De nombreuses perspectives sont ainsi envisageables pour améliorer la scalabilité de la méthode et passer à des centaines de millions ou des milliards de points d'une part, et pour mieux traiter des données bruitées d'autre part.

Les performances de notre algorithme de reconstruction sélective pourraient être améliorées de façon significative en réduisant le nombre de requêtes spatiales. Une piste en cours d'étude consisterait à relaxer localement la propriété de Delaunay globale en autorisant le choix de candidats approximatifs pour répondre à la requête Q_{dt} et en réparant les erreurs à la volée lorsque cela est nécessaire par mise à jour locale. Des statistiques montrent en effet que les positions des sommets frontaux dans la triangulation permettent souvent de former des tétraèdres de Delaunay, ce qui en fait des candidats à privilégier.

Nous étudions également une manière d'améliorer la scalabilité de notre approche dynamique en écrivant le résultat en mémoire externe sous la forme d'un flux cohérent de triangles à la manière d'Isenburg et al. [115]. Ce type d'approche permet en outre de transmettre le résultat à des algorithmes de traitement de maillages s'appliquant à des flux de donnée, donnant lieu à une chaîne de traitement particulièrement efficace pour un faible coût en mémoire.

Enfin, nous souhaitons étendre l'algorithme de reconstruction sélective au rééchantillonnage afin de pouvoir produire des échantillons de meilleure qualité à partir de données bruitées ou incomplètes, ou à partir d'images de profondeur mal recalées. Une solution possible consisterait à utiliser une représentation implicite locale pour placer les points de l'échantillon final de manière optimale sur la surface, comme proposé par Ohtake et al. [156], en ajoutant également des points dans les zones où la surface de l'objet à reconstruire a été mal échantillonnée. Plus que réduire la redondance des données, il serait aussi intéressant de pouvoir simplifier directement la forme d'une surface selon une technique de rééchantillonnage à la volée, en contrôlant le niveau de détail de la géométrie vis à vis d'une échelle.

D'autres questions intéressantes pour l'avenir, soulevées par ces travaux, concernent les garanties sur l'échantillon simplifié et l'approximation produite par l'algorithme de reconstruction sélective, les conditions d'échantillonnage sous lesquelles la convection géométrique produit une surface homéomorphe à l'originale, et l'extension de l'ensemble de ces résultats à des surfaces non lisses.

Annexes

de la première partie

Triangulation de Delaunay

Dans cette annexe, nous commençons par rappeler succinctement quelques définitions et propriétés relatives aux triangulations de Delaunay d'un ensemble de points de \mathbb{E}^d . Nous rappelons ensuite les notions d'axe médian et d' ε -échantillon d'une surface plongée dans \mathbb{R}^3 , puis celle de triangulation de Delaunay restreinte. Pour plus de détails, le lecteur pourra se reporter à l'ouvrage de Boissonnat et Yvinec [46] ou à l'article de Cazals et Giesen [51].

A.1 Triangulations de Delaunay

Soit P un ensemble fini de points d'un espace euclidien \mathbb{E}^d .

A.1.1 Diagramme de Voronoï

On appelle *cellule de Voronoï* d'un point $\mathbf{p} \in P$, et on note $V(\mathbf{p})$, l'ensemble des points de \mathbb{E}^d plus proches de \mathbf{p} que tout autre point de P , i.e. :

$$V(\mathbf{p}) = \{\mathbf{x} \in \mathbb{E}^d \mid \forall \mathbf{q} \in P, \|\mathbf{x} - \mathbf{p}\| \leq \|\mathbf{x} - \mathbf{q}\|\}$$

La cellule de Voronoï $V(\mathbf{p})$ est un polyèdre convexe. Le *diagramme de Voronoï* de P l'ensemble des cellules de Voronoï des points de P et de leur *faces*, i.e. l'ensemble des points communs à k cellules de Voronoï, avec $1 \leq k \leq d$. Les faces partagées par deux cellules de Voronoï sont appelées *facettes de Voronoï*, les faces partagées par trois cellules de Voronoï sont appelées *arêtes de Voronoï* et les faces partagées par quatre cellules de Voronoï sont appelés *sommets de Voronoï*.

A.1.2 Triangulation de Delaunay

On appelle *simplexe* de dimension d l'enveloppe convexe de d points dans \mathbb{E}^d (arête pour $d = 1$, triangle pour $d = 2$, tétraèdre pour $d = 3$). On appelle *simplexe de Delaunay* de P tout simplexe de sommets $\mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_k$ de dimension $k \leq d$, $\mathbf{p}_i \in P$, tel qu'il existe une boule passant par $\mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_k$ ne contenant pas de points de P en son intérieur. On appelle *triangulation de Delaunay* de P , et on note $D(P)$, le complexe formé des simplexes de Delaunay de P .

La triangulation de Delaunay $D(P)$ est le dual du diagramme de Voronoï dans le sens où un simplexe $\mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_k$ appartient à $D(P)$ si et seulement si les cellules de Voronoï associées aux points $\mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_k$ ont une face de Voronoï en commun. Cette triangulation est un complexe simplicial qui décompose l'enveloppe convexe de P . Un simplexe formé de quatre points de P définit une *cellule de Delaunay* (tétraèdre) si les cellules de Voronoï de ces points ont un sommet en commun. Un simplexe formé de trois points de P est une *facette de Delaunay* (triangle) si leurs cellules de Voronoï partagent une arête. Enfin, tout point de P est associé à un *sommet de Delaunay*.

La triangulation de Delaunay est unique si les points sont en position générale, i.e. s'il n'existe pas $d + 2$ points cosphériques dans P . Dans les cas où de telles configurations apparaissent, le recours à des schémas de perturbation permet d'obtenir un résultat fixe [87].

La triangulation de Delaunay possède de nombreuses propriétés qui expliquent son utilisation fréquente en modélisation géométrique, notamment en reconstruction de surface. En particulier, elle maximise le plus petit des angles de ses triangles. Une autre propriété utile est que ses facettes extérieures forment la frontière de l'enveloppe convexe de l'ensemble de points triangulé. La figure A.1 montre la triangulation de Delaunay et le diagramme de Voronoï d'un échantillon de points dans le plan.

Complexité La triangulation de Delaunay 3D peut être calculée de manière incrémentale en temps $O(N \log N)$, où N est le nombre de points de l'échantillon en utilisant une structure hiérarchique [71]. Pour un ε -échantillon localement uniforme, le nombre de cellules de la triangulation de Delaunay 3D est presque linéaire par rapport à N [22].

Mise à jour L'insertion d'un point \mathbf{x} dans une triangulation de Delaunay est une tâche simple qui s'effectue en deux étapes. Tout d'abord, on cherche les cellules en conflit, i.e. les tétraèdres dont la sphère circonscrite contient \mathbf{x} . L'union de ces cellules forme une région connexe appelée *cavité*. Cette cavité est ensuite retriangulée avec de nouvelles cellules. Chacune de ces cellules est construite à partir d'une facette de Delaunay du bord de la cavité et du nouveau sommet inséré. La localisation de la première cellule en conflit est la tâche la plus coûteuse, qui s'effectue en moyenne en temps $O(\log N)$, où N est le nombre de points de l'échantillon, avec l'utilisation d'une hiérarchie de Delaunay [71]. L'insertion de plusieurs points successivement de manière localisée peut être optimisée en choisissant pour chaque nouveau point une cellule de départ parmi les cellules créées suite à l'insertion du point précédent. Lorsqu'un point est retiré, une cavité est également créée, qu'il suffit de retriangler à partir des sommets du bord de la cavité pour mettre à jour la triangulation.

A.1.3 Pôles

Soit $V_{\mathbf{p}}$ la cellule de Voronoï d'un point \mathbf{p} d'un ensemble P . Alors le pôle positif \mathbf{p}^+ est défini comme le sommet de Voronoï de $V_{\mathbf{p}}$ le plus éloigné de \mathbf{p} . Le vecteur \mathbf{pp}^+ , appelé *vecteur pôle*, fournit une approximation de la direction de la normale en \mathbf{p} . Le pôle négatif \mathbf{p}^- est défini comme le sommet de la cellule de Voronoï $V_{\mathbf{p}}$ le plus éloigné de \mathbf{p}^+ dans le sens opposé au vecteur \mathbf{pp}^+ . Les pôles fournissent

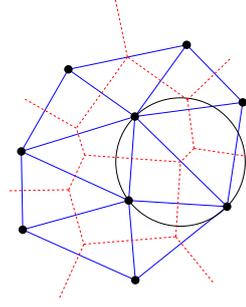


FIG. A.1 – Diagramme de Voronoï (arêtes en pointillés) et triangulation de Delaunay (arêtes solides) d'un ensemble de points dans le plan.

une bonne approximation de l'axe médian de la surface échantillonnée par P . Sous l'hypothèse d'un ε -échantillon d'une surface lisse plongée dans \mathbb{R}^3 (notion définie plus bas), Amenta et Bern [13] ont prouvé la convergence des normales estimées à partir des pôles vers celles de la surface échantillonnée, et Boissonnat et Cazals [39] d'une part et Amenta et al. [16] d'autre part ont démontré la convergence des pôles vers l'axe médian pour $\varepsilon \rightarrow 0$.

A.2 Axe médian et ε -échantillon

Soit S une surface lisse plongée dans \mathbb{R}^3 et $P \subset S$ un échantillon fini de points de S .

A.2.1 Axe médian

Une boule dans \mathbb{R}^3 est dite vide si elle ne contient aucun point de S en son intérieur. Une boule vide est dite maximale si elle n'est contenue dans aucune autre boule vide. L'axe médian $M(S)$ de S est l'ensemble de tous les centres des boules maximales vides dans \mathbb{R}^3 . Les boules maximales vides sont tangentes en au moins deux points de S .

La transformation qui associe à une forme son axe médian n'est pas continue. Une petite perturbation dans la forme peut entraîner une perturbation importante de l'axe médian. Cet objet est donc difficile à calculer. On peut cependant approximer l'axe médian d'une surface à partir du diagramme de Voronoï d'un échantillon de points sur cette surface de manière stable [21, 53]. La figure A.2 montre l'axe médian d'une courbe plane.

A.2.2 Local feature size

Le *local feature size* $\text{lfs}(\mathbf{x})$ d'un point $\mathbf{x} \in S$ est une fonction $\text{lfs} : S \rightarrow \mathbb{R}$ qui mesure la distance minimale entre \mathbf{x} et l'axe médian $M(S)$, i.e. :

$$\text{lfs}(\mathbf{x}) = \min_{\mathbf{y} \in M(S)} \|\mathbf{x} - \mathbf{y}\|$$

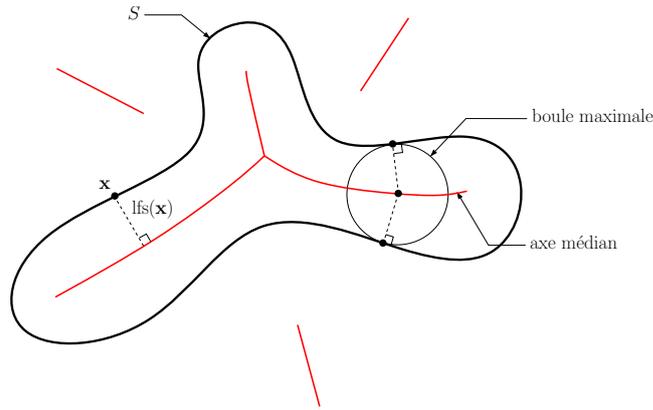


FIG. A.2 – Axe médian d’une courbe S plane.

Amenta et Bern [13] ont montré que le local feature size est une fonction continue 1-lipschitzienne, i.e. $\text{lfs}(\mathbf{x}) \leq \text{lfs}(\mathbf{y}) + \|\mathbf{x} - \mathbf{y}\|$ pour tous $\mathbf{x}, \mathbf{y} \in S$.

A.2.3 Conditions d’échantillonnage

Les conditions d’échantillonnage permettent de déduire des propriétés de la surface à partir de l’échantillon de points, et en particulier d’estimer les normales.

ε -échantillon Un échantillon de points P est un ε -échantillon d’une surface S si pour tout point $\mathbf{x} \in S$ il existe un point $\mathbf{p} \in P$ situé à une distance au plus $\varepsilon \text{lfs}(\mathbf{x})$, i.e. $\|\mathbf{x} - \mathbf{p}\| \leq \varepsilon \text{lfs}(\mathbf{x})$.

Le critère d’ ε -échantillon garantit qu’une surface S est échantillonnée de manière dense dans les régions de forte courbure et où deux nappes séparées de la surface se rapprochent. Dey et al. [76] d’une part, et Funke et Ramos [94] d’autre part ont introduit la notion d’ ε -échantillon localement uniforme.

(ε, δ) -échantillon Un échantillon de points P est appelé (ε, δ) -échantillon d’une surface S pour $\frac{\varepsilon}{2} \leq \delta < \varepsilon \leq 1$ si P est un ε -échantillon de S et que $\|\mathbf{p} - \mathbf{q}\| \geq \delta \text{lfs}(\mathbf{p})$ pour tous $\mathbf{p}, \mathbf{q} \in P$.

Le critère d’ ε -échantillon donne une borne inférieure sur la densité de l’échantillon, tandis que le critère s’appuyant sur δ fournit une borne supérieure et contrôle la position des points de l’échantillon pour assurer une certaine uniformité de l’échantillonnage.

A.3 Triangulation de Delaunay restreinte à une surface

La notion de triangulation de Delaunay restreinte à une surface a été introduite par Chew [57] dans le cadre d’un algorithme d’échantillonnage de surface, puis formalisée par Edelsbrunner et Shah [89], qui en ont étudié les propriétés. En particulier, ils ont montré que la triangulation de Delaunay restreinte est homéomorphe à la surface sous certaines conditions d’échantillonnage.

On appelle triangulation de Delaunay de P restreinte à S , et on note $D_S(P)$ l'ensemble des faces de Delaunay dont les faces de Voronoï duales intersectent S (Fig. A.3).

Edelsbrunner et Shah ont montré que $D_S(P)$ est homéomorphe à S si chaque cellule de Voronoï satisfait une condition appelée propriété de boule fermée (*closed ball property*). Amenta et al. [14] ont montré que si P est un ε -échantillon, avec $\varepsilon \leq 0.08$, alors la triangulation de Delaunay restreinte $D_S(P)$ est une surface linéaire par morceaux homéomorphe à S .

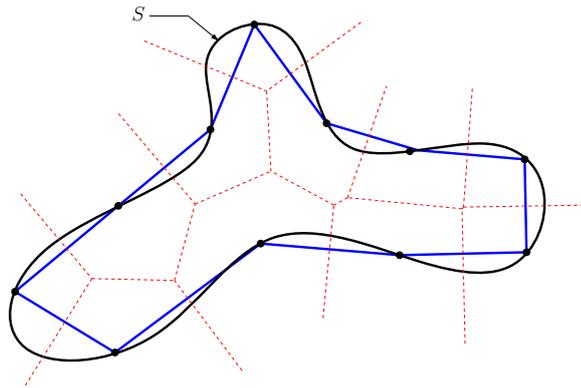


FIG. A.3 – Triangulation de Delaunay restreinte d'une courbe plane.

Estimation locale de la direction des vecteurs normaux

Lorsque les vecteurs normaux en chaque point ne sont pas fournis par les dispositifs d'acquisition ou bien que leur mesure n'est pas suffisamment fiable, il peut être nécessaire de les estimer. Une première solution consiste à les estimer à partir des pôles du diagramme de Voronoï des points de l'échantillon (voir l'Annexe A), avec une garantie de convergence pour un ε -échantillon avec $\varepsilon \rightarrow 0$. Une autre approche courante consiste à estimer localement le plan tangent en chaque point en considérant un voisinage local. En s'appuyant sur les relations de voisinage locales, les propriétés locales de la surface en un point $\mathbf{p} \in P$ peuvent être estimées par analyse statistique de la distribution spatiale des points voisins. En particulier, l'analyse en composantes principales de la matrice de covariance de la position des points dans un voisinage local fournit un algorithme efficace pour estimer les vecteurs normaux.

Soit $\bar{\mathbf{p}}$ l'isobarycentre du voisinage de \mathbf{p} , i.e.,

$$\bar{\mathbf{p}} = \frac{1}{k} \sum_{i=1}^k \mathbf{p}_i$$

La matrice de covariance C pour \mathbf{p} est la matrice 3×3 donnée par :

$$C = \sum_{i=1}^k (\mathbf{p}_i - \bar{\mathbf{p}})^T (\mathbf{p}_i - \bar{\mathbf{p}})$$

La matrice C décrit les propriétés statistiques de distribution des points dans le voisinage de \mathbf{p} en sommant les distances au carré de ces points au barycentre $\bar{\mathbf{p}}$. La direction associée à la plus petite valeur propre de C donne la direction de la normale à un plan tangent en \mathbf{p} .

La fiabilité de cette estimation dépend clairement des conditions d'échantillonnage. Pour un ensemble de points qui est un ε -échantillon d'une surface lisse, Amenta et Bern [13] donnent une borne supérieure sur l'erreur entre des normales estimées à partir des pôles et les normales de la vraie surface. Ce résultat ne s'applique pas pour des normales estimées à partir des k plus proches voisins, à cause de la sensibilité inhérente de cette approche à la distribution des points et au bruit. Pour rendre l'estimation plus robuste, différentes stratégies sont possibles. Linsen [130] a proposé d'examiner l'angle entre les arêtes formées par \mathbf{p} et ses plus proches voisins successifs. Mitra et Nguyen [143] ont proposé

une procédure de raffinement itératif pour déterminer la valeur de k en chaque point, en tenant compte d'informations relatives au capteur.

Sous l'hypothèse que l'ensemble de points est *localement uniforme*, Andersson et al. [18] ont établi une borne inférieure et une borne supérieure sur la valeur de k de manière à ce que tous les voisins d'un point \mathbf{p} dans le Delaunay restreint sont contenus dans les k plus proches voisins de \mathbf{p} . Cela permet une estimation fiable de la normale en \mathbf{p} . Cependant, les échantillons de points issus de la numérisation d'objets physiques ne sont pas nécessairement localement uniforme, en particulier à l'issue d'une phase de fusion de plusieurs images de profondeur. Si l'échantillon correspondant à chaque image peut vérifier une condition d'uniformité locale résultant du processus d'acquisition, il n'en va pas de même lorsque l'on considère un échantillon dans lequel plusieurs images de profondeur se recouvrent et mènent à une redondance des données localement. Les techniques de fusion d'images de profondeur telles que celles de Curless et Levoy [67] ou Rocchini et al. [164] sont cependant capables de fournir des estimations fiables des normales à partir de celles des images de profondeur. Dans le cas où les vecteurs normaux d'un échantillon de points sans bonnes propriétés de distribution doivent être estimés de manière fiable, le recours aux pôles, voire à une approche interactive peut s'avérer nécessaire.

On peut noter enfin que si une orientation globale cohérente des vecteurs normaux est nécessaire, ce qui n'est pas le cas dans notre méthode, le recours à des heuristiques s'impose, comme par exemple la propagation de l'orientation le long d'un arbre minimum recouvrant euclidien [112].

Deuxième partie

Modélisation géométrique multireprésentation

Dans le chapitre 1, nous proposons quelques rappels sur les deux modèles impliqués dans notre système de modélisation : les surfaces implicites et les maillages polygonaux, et présentons un état de l'art des opérations de combinaison et de déformation associées. Le chapitre 2 présente une étude bibliographique des méthodes multireprésentations combinant ces modèles. Dans le chapitre 3, nous décrivons l'HybridTree de façon détaillée et présentons quelques résultats expérimentaux, avant de conclure dans le chapitre 4.

Surfaces implicites et maillages polygonaux

Dans ce chapitre, nous rappelons quelques définitions et propriétés de base sur les surfaces implicites et les maillages polygonaux, ainsi que l'état de l'art des principaux outils de combinaison et de déformation associés en soulignant les avantages et les limites de chaque modèle.

1.1 Surfaces implicites

1.1.1 Définition et propriétés

Une surface implicite S est définie de manière générale par une fonction potentiel $f : \mathbb{R}^3 \rightarrow \mathbb{R}$ qui associe une valeur scalaire à tout point de l'espace. La surface S est alors définie comme l'ensemble des points auxquels la fonction f s'annule, i.e. :

$$S = \{\mathbf{p} \in \mathbb{R}^3 \mid f(\mathbf{p}) = 0\}.$$

Si la surface S est fermée, alors la fonction f permet de séparer l'espace en deux composantes, l'intérieur (le *volume*) et l'extérieur de la surface S . On obtient ainsi une classification de tout point $\mathbf{p} \in \mathbb{R}^3$ par rapport à S :

- Si $f(\mathbf{p}) > 0$, alors \mathbf{p} est à l'intérieur de la surface.
- Si $f(\mathbf{p}) = 0$, alors \mathbf{p} est sur la surface.
- Si $f(\mathbf{p}) < 0$, alors \mathbf{p} est à l'extérieur de la surface.

La convention opposée peut aussi être utilisée pour définir l'intérieur et l'extérieur de la surface, i.e. $f(\mathbf{p}) < 0$ pour l'intérieur, et $f(\mathbf{p}) > 0$ pour l'extérieur.

Si f est une fonction continue et différentiable, la normale en un point de la surface peut être calculée de manière analytique à partir du gradient ∇f de f , défini comme suit :

$$\nabla f(\mathbf{p}) = \left[\frac{\partial f}{\partial x}(\mathbf{p}), \frac{\partial f}{\partial y}(\mathbf{p}), \frac{\partial f}{\partial z}(\mathbf{p}) \right].$$

Une propriété remarquable des surfaces implicites est qu'il est possible de combiner plusieurs fonctions potentiel indépendantes pour générer une fonction potentiel globale. On peut ainsi représenter des

formes complexes par combinaison ou transformation de primitives définies par des fonctions simples f_1, f_2, \dots, f_n , avec des opérations dont le résultat est une nouvelle surface implicite.

Il existe principalement deux types de formulations pour décrire des formes simples de manière implicite : la formulation analytique et la formulation par distance à un squelette (section 1.1.2). Ces surfaces peuvent être facilement combinées par des opérations booléennes ou de mélange, et déformées à l'aide de transformations de l'espace. Ces opérations sont généralement organisées dans une structure hiérarchique de type arbre CSG (*Constructive Solid Geometry*), où les feuilles correspondent aux primitives et les nœuds aux opérations (sections 1.1.3 et 1.1.4). Dans notre système, nous utilisons le modèle des *surfaces implicites à squelettes* qui sera présenté plus en détails dans la section 1.1.5. Nous évoquerons enfin les représentations implicites discrètes dans la section 1.1.6.

1.1.2 Primitives implicites

Des primitives implicites peuvent être définies principalement de deux manières : analytiquement ou à partir d'une fonction de distance à un squelette géométrique.

Primitives analytiques Des formes simples comme des plans, des sphères, des ellipsoïdes, des cylindres, des cônes ou des tores peuvent être définis de manière implicite par des polynômes. Les superellipsoïdes [193] et les superquadriques [28] sont des classes de surfaces algébriques qui permettent de représenter un certain éventail de formes. Par exemple, les superquadriques se formulent ainsi :

$$f(\mathbf{p}) = \frac{\mathbf{p}_x^k}{a^k} + \frac{\mathbf{p}_y^k}{b^k} + \frac{\mathbf{p}_z^k}{c^k}.$$

De manière générale, la description analytique d'une forme géométrique est fastidieuse et manque de paramètres de contrôle intuitifs pour représenter des formes réalistes.

Primitives à squelette Des primitives implicites plus simples à manipuler peuvent être construites à partir d'objets géométriques appelés *squelettes* (ou parfois *éléments structurants*), générant un champ de potentiel dans l'espace défini par une fonction de la forme :

$$f(\mathbf{p}) = C - d(\mathbf{p})$$

où $d : \mathbb{R}^3 \rightarrow \mathbb{R}_+$ est une fonction de distance au squelette, et $C > 0$ est une constante. Les surfaces implicites à squelettes ont été introduites pour la première fois par par Blinn [36], qui a proposé d'utiliser des points comme squelettes, mais des formes plus complexes peuvent être manipulées. Cette approche permet ainsi de créer des formes de base facilement contrôlables pour l'utilisateur. Les primitives à squelette constituent la base du modèle des surfaces implicites à squelette que nous présenterons en détails dans la section 1.1.5.

1.1.3 Opérations booléennes et mélange

Les surfaces implicites peuvent être aisément combinées par les opérations CSG classiques d'union, d'intersection et de différence. Ces opérations ont été généralisées par des opérations de mélange qui permettent de générer des raccords lisses entre les primitives.

Les fonctions min et max proposées par Ricci [163] constituent le moyen le plus simple pour définir des opérations booléennes entre deux surfaces implicites :

$$\begin{aligned} \text{Union} & : f_1 \cup f_2 = \max(f_A, f_B) \\ \text{Intersection} & : f_1 \cap f_2 = \min(f_A, f_B) \\ \text{Différence} & : f_1 \setminus f_2 = f_1 \cap (-f_2) \end{aligned}$$

Ces formules introduisent des discontinuités dans le champ de potentiel. Différentes formulations ont été proposées pour les éviter et produire des raccords lisses entre les surfaces. Une solution consiste à utiliser la formulation par R-fonctions proposée par Pasko et al. [157] :

$$\begin{aligned} f_1 \cup_\alpha f_2 &= \frac{1}{1+\alpha} (f_1 + f_2 + \sqrt{f_1^2 + f_2^2 - 2\alpha f_1 f_2}) \\ f_1 \cap_\alpha f_2 &= \frac{1}{1+\alpha} (f_1 + f_2 - \sqrt{f_1^2 + f_2^2 - 2\alpha f_1 f_2}) \end{aligned}$$

Le paramètre $\alpha \in [0, 1]$ dont dépendent ces fonctions produit un raccordement lisse pour $\alpha = 0$, et celles-ci convergent vers les fonctions min et max classiques lorsque α tend vers 1. Ces fonctions définissent un champ de potentiel avec continuité C^1 partout dans l'espace, excepté sur la surface. Pasko et al. [157] ont également proposé des fonctions de continuité C^m définies comme suit :

$$\begin{aligned} f_1 \cup_m f_2 &= (f_1 + f_2 + \sqrt{f_1^2 + f_2^2})(f_1^2 + f_2^2)^{\frac{m}{2}} \\ f_1 \cap_m f_2 &= (f_1 + f_2 - \sqrt{f_1^2 + f_2^2})(f_1^2 + f_2^2)^{\frac{m}{2}} \end{aligned}$$

Une autre famille de fonctions C^n permettant de représenter des opérations booléennes et de mélange fondées sur l'opérateur superelliptique de Ricci [163] a été proposée par Wyvill et al. [191] :

$$f_1 \cup_n f_2 = (f_1^{-n} + f_2^{-n})^{-\frac{1}{n}} \quad f_1 \cap_n f_2 = (f_1^n + f_2^n)^{\frac{1}{n}}$$

Pour cette famille de fonctions, le paramètre n permet de retrouver les opérations d'union et d'intersection lorsque n tend vers $+\infty$.

Pour un meilleur contrôle de la forme de la transition, de nombreuses autres fonctions ont été étudiées

dans la littérature. Par exemple, Pasko et al. [157] ont proposé la formulation suivante :

$$f = f_1 + f_2 + \sqrt{f_1^2 + f_2^2} + \frac{a_0}{1 + \left(\frac{f_1}{a_1}\right)^2 + \left(\frac{f_2}{a_2}\right)^2}$$

où les paramètres a_0 , a_1 et a_2 permettent de contrôler la forme de la transition. Barthe et al. [30] définissent la forme de la transition de manière interactive à l'aide de courbes spline.

La définition globale du mélange peut conduire à la formation locale de raccords indésirables entre deux surfaces implicites. Pour y remédier, les interactions entre les primitives peuvent être décrites par un graphe de mélange, comme par exemple proposé par Cani et al. [49]. Cette solution résulte en une union de volumes pouvant créer des discontinuités tangentielles dans les régions où les propriétés de mélange changent. Des opérations de mélange local moins contraignantes ont été développées par Pasko et al. [158]. Plusieurs régions de mélange local entre deux surfaces peuvent être définies, chacune limitée par un volume englobant prédéfini.

1.1.4 Déformations

La fonction potentiel f_w d'une surface implicite déformée par une transformation $w : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ peut s'exprimer sous la forme :

$$f_w(\mathbf{p}) = f \circ w^{-1}(\mathbf{p})$$

où f est la fonction définissant la surface originale et w^{-1} représente la transformation inverse. La fonction f_w n'est donc valide que si la transformation w^{-1} est définie. Les transformations affines et la famille des déformations axiales, introduites par Barr [29] et généralisées par Blanc [34], font partie des déformations inversibles. Cependant, elles sont globales et limitées dans les formes qu'elles permettent de produire. Les techniques plus locales de déformation telles que celles de la famille des FFD (*Free-Form Deformations*) [174], développées pour les maillages polygonaux ou les surfaces paramétriques, n'offrent pas d'expression analytique de la déformation inverse w^{-1} . Elles ne s'appliquent ainsi pas directement aux surfaces implicites.

1.1.5 Modèle des surfaces implicites à squelettes

Les surfaces implicites à squelettes ont été introduits en synthèse d'images au début des années 1980 avec les travaux de Blinn [36] (*blobs*), Nishimura et al. [150] (*metaballs*) puis Wyvill et al. [194] (*soft objects*). Ces modèles simplifient la création de formes complexes réalistes à partir de surfaces implicites et ont contribué à leur démocratisation.

Les modèles à squelettes représentent un objet par un ensemble de primitives à squelette générant chacune un champ de potentiel dans l'espace défini par une fonction de la forme :

$$f_i(\mathbf{p}) = g_i \circ d_i(\mathbf{p})$$

où $d_i : \mathbb{R}^3 \rightarrow \mathbb{R}_+$ est une fonction de distance au squelette, et $g_i : \mathbb{R}_+ \rightarrow \mathbb{R}$ une fonction de distribution du potentiel dans l'espace, dont la valeur décroît en fonction de la distance au squelette.

Dans cette formulation, les primitives à squelette définissent à la fois leur forme et la façon dont elles peuvent se mélanger. Ainsi, la fonction potentiel f qui résulte du mélange de n primitives peut se formuler très simplement :

$$f(\mathbf{p}) = \sum_{i=1}^n f_i(\mathbf{p})$$

Une surface implicite à squelette est alors définie comme une *isosurface* pour une valeur de seuil $T > 0$ fixée :

$$S = \{\mathbf{p} \in \mathbb{R}^3 \mid f(\mathbf{p}) = T\}$$

Le volume de la surface S est alors caractérisé par l'ensemble des points $\mathbf{p} \in \mathbb{R}^3$ tels que $f(\mathbf{p}) > 0$.

Dans le modèle original de Blinn [36], les squelettes sont simplement des points, mais des formes plus complexes peuvent être utilisées, comme des segments, des polygones, des courbes, des surfaces paramétriques, ou des polytopes [191, 27]. La fonction de distance d_i utilisée est en général la distance euclidienne minimale au squelette, c'est-à-dire :

$$d_i(\mathbf{p}) = \min_{\mathbf{x} \in \mathcal{S}_i} \|\mathbf{x} - \mathbf{p}\|$$

Si \mathcal{S}_i est une forme géométrique simple, cette distance peut facilement être calculée de manière analytique. Pour des formes plus complexes, le calcul peut nécessiter des algorithmes sophistiqués [173, 26].

La fonction de distance à un squelette peut également être une distance anisotrope, dont la valeur dépend de la direction autour du squelette. Diverses formes peuvent ainsi être obtenues à partir d'un squelette associé à la distance \mathcal{L}^p , comme proposé par Blanc et Schlick dans [35]. La fonction de distance fait alors intervenir un paramètre p permettant de contrôler la forme de la surface :

$$d_i(\mathbf{p}) = (|\mathbf{p}_x|^p + |\mathbf{p}_y|^p + |\mathbf{p}_z|^p)^{\frac{1}{p}} \quad p \in]0, +\infty[$$

D'autres fonctions de distance anisotrope peuvent être définies à partir de polyèdres frontières étoilés, comme proposé par Singh et Parent [177]. Crespin et al. [66] ont défini des primitives implicites par révolution et balayage d'un profil le long d'une trajectoire.

La fonction g_i de distribution de potentiel est en général une gaussienne, ou bien une approximation polynomiale. Différentes fonctions à support fini approximant la gaussienne ont été proposées dans la littérature, permettant de réduire le coût d'évaluation de la fonction potentiel pour des modèles complexes. Ces fonctions atteignent un maximum en 0, i.e. au niveau du squelette, et apportent une contribution lorsque la distance $r = d_i(\mathbf{p})$ entre un point $\mathbf{p} \in \mathbb{R}^3$ et le squelette est inférieure à un rayon d'influence fixé R , et nulle lorsque r est supérieur à R . Pour assurer un mélange continu, ces fonctions doivent vérifier certaines propriétés de continuité, en particulier $g'_i(0) = 0$ et $g'_i(R) = 0$.

En s'appuyant sur cette représentation, Wyvill et al. [191] ont développé le modèle hiérarchique

du BlobTree, qui est un arbre CSG étendu combinant des primitives à squelette à l'aide d'opérations booléennes, de mélange et de déformation. Le modèle hybride que nous proposons dans cette thèse est une généralisation du BlobTree manipulant des représentations implicites et maillées.

Un inconvénient des surfaces implicites à squelette est l'apparition de boursoufflures au niveau des zones de mélange dans les régions où les volumes de plusieurs primitives se recouvrent. Les surfaces de convolution introduites par Bloomenthal et Shoemake [37] sont une évolution des surfaces implicites à squelettes permettant de remédier à ce défaut. Le mélange n'est plus effectué en sommant les fonctions potentiel associées aux primitives mais leur intégrale sur l'union des squelettes. Ce dernier modèle a notamment été appliqué par Angelidis et Cani à des squelettes construits à partir de réseaux de courbes ou de surfaces de subdivision [19].

1.1.6 Représentations discrètes

Plusieurs représentations implicites discrètes manipulant un échantillonnage de la fonction potentiel sur une grille 3D ont été développées. L'évaluation de la fonction potentiel ne dépend alors plus de la complexité de l'objet. Celle-ci peut-être réalisée par propagation locale du potentiel, en stockant le potentiel seulement dans un voisinage autour de la surface.

Museth et al. [148] ont proposé un modèle polyvalent pour l'édition de formes complexes reposant sur un modèle de surface implicite dynamique et la méthode des level sets. Le système fournit des algorithmes de conversion à partir de nombreuses autres représentations, incluant les nuages de points et les maillages polygonaux, et une panoplie importante d'outils d'édition. Cette approche utilise une grille régulière, ce qui limite la précision des résultats obtenus. Dans la même lignée, des systèmes de sculpture virtuelle reposant sur une représentation implicite échantillonnée sur une grille ont également été développés. Le système de Ferley et al. utilise un modèle de blobs dont le potentiel est échantillonné sur une grille régulière [90]. Le modèle des *Adaptively Sampled Distance Fields* introduits par Frisken et al. [93] s'appuie sur une structure hiérarchique permettant un contrôle local sur l'erreur géométrique. Cette représentation est moins coûteuse qu'une grille régulière, et permet la gestion naturelle de niveaux de détails. Les outils de sculpture proposés se limitent à des opérations booléennes locales. Les arêtes vives qui en résultent sont bien représentées par la gestion des niveaux de détail. Les mises à jour du potentiel lors des opérations peuvent nécessiter de mettre à jour la densité de l'échantillon, ce qui peut constituer une opération coûteuse.

1.2 Maillages polygonaux

1.2.1 Définition et propriétés

Rappelons qu'un maillage polygonal, auquel nous ferons référence simplement par le terme *maillage*, est un complexe linéaire par morceaux représentant une surface sous la forme d'un réseau cohérent de

sommets, arêtes et faces. Un maillage définit la géométrie tout en donnant une topologie de la surface qu'il représente, décrit par l'intermédiaire de relations de connectivité entre les sommets.

Les maillages représentant des objets réalistes ont en général une topologie de 2-variété, souvent appelés maillages *manifolds*. Un maillage est *manifold* s'il satisfait la propriété locale de disque topologique suivante : en tout point du maillage, il existe une sphère de rayon $\varepsilon > 0$ telle que l'intersection entre la sphère et le maillage est homéomorphe à un disque ou à un demi-disque (cas des surfaces ouvertes).

Il existe une grande diversité de structures de données pour représenter des maillages. Le choix d'une structure particulière est guidé par l'utilisation, et les algorithmes que l'on souhaite appliquer. Certaines structures privilégient la compacité, telles que les bandes de triangles, par exemple dans le but d'accélérer la visualisation. D'autres structures comme les arêtes ailées [31], les demi-arêtes [135], ou les arêtes orientées [48] sont optimisées pour le parcours de la surface, ce qui constitue une opération fréquente dans les algorithmes géométriques.

La manipulation de maillages requiert de gérer de façon cohérente deux types d'informations complémentaires : l'information géométrique relative à la position des sommets, et l'informations topologiques, relative à leur connectivité. Les opérations affectant uniquement la géométrie, telles que les transformations affines et certaines déformations, peuvent s'appliquer de façon intuitive et immédiate (section 1.2.2). En revanche, les opérations de combinaison booléennes ou les opérations de type mélange sont plus complexes à mettre en œuvre sur cette représentation (section 1.2.3). Dans les développements récents figurent les représentations différentielles, qui sont intéressantes pour contrôler des opérations de déformation et de mélange (section 1.2.4).

1.2.2 Déformations

Un grand nombre de méthodes de déformation de surfaces ont été développées depuis une vingtaine d'années [32]. A l'origine de ces techniques, Barr [29] propose des transformations globales et locales pour des modèles maillés. Des formules mathématiques définissent explicitement les transformations à appliquer aux points de l'objet à déformer. Les techniques de déformation de l'espace sont indépendantes de la représentation utilisée. Elles définissent une déformation par une fonction continue de \mathbb{R}^3 dans \mathbb{R}^3 , déformant l'espace ou une zone de l'espace dans lequel est plongé l'objet. Sederberg et Parry [174] sont à l'origine des techniques de FFD. Elles modélisent une déformation à l'aide d'une grille régulière de points de contrôle dans l'espace. Les déplacements exercés au niveau des points de contrôle sont repercutés sur les objets plongés dans la grille. Des extensions de ce modèle permettent d'utiliser des grilles de formes plus variées [63] et de topologies différentes [134]. Cependant, ces techniques requièrent la manipulation d'un grand nombre de points de contrôle pour créer des formes complexes.

Borrel [47] a proposé une méthode où les déformations de l'espace sont définies par contraintes. Chaque déformation élémentaire est composée d'un ou plusieurs points de contrôle et d'un volume d'influence caractérisant l'aspect local ou global de la transformation. Les contraintes de déplacement exercées sur les points sont repercutées dans la limite du volume d'influence.

La localisation immédiate de points sur une surface maillée rend ainsi les maillages particulièrement bien adaptés aux déformations, qui s'appliquent de façon intuitive et interactive. Des précautions particulières doivent cependant être prises pour éviter des incohérences topologiques du type auto-intersection.

1.2.3 Opérations booléennes et mélange

Les opérations booléennes sur les maillages sont importantes dans de nombreux domaines d'application de la modélisation géométrique, comme en CAO par exemple. Cependant, la représentation des objets par leur frontière rend ces opérations très complexes. La principale difficulté est d'assurer la cohérence topologique du résultat : une opération appliquée à deux solides doit produire un autre solide cohérent. Des problèmes se posent alors quant à la définition exacte et complète des opérations, et quant à la robustesse numérique de leur évaluation. Ces opérations ont été longuement étudiées en géométrie algorithmique [54, 84, 97], et font toujours l'objet de recherches actives, notamment avec une étude récente réalisées sur des polyèdres de Nef [108].

Les maillages n'offrent pas de définition naturelle de l'opération de raccordement lisse dans un sens volumétrique. Il est en revanche possible de déformer localement les surfaces à fusionner pour aboutir à des formes similaires à celles que l'on pourrait obtenir dans un cadre implicite, une difficulté étant toujours d'assurer la cohérence topologique du résultat. Plusieurs techniques permettant ce type de fusion ont été étudiées ces dernières années, avec notamment un fort engouement pour des représentations par coordonnées différentielles, qui seront présentées dans la section suivante.

Interpolation géométrique locale Il est possible de fusionner directement deux modèles maillés en mettant en correspondance les régions qui doivent se raccorder, puis en déformant localement les surfaces pour assurer une transition lisse. Kanai et al. [118] ont proposé l'une des premières méthodes procédant selon ce principe. Après sélection des parties d'intérêt sur différents maillages, une correspondance entre les sommets des bords est d'abord établie. Ensuite, un processus de recalage est appliqué entre les deux bords et une fonction B-spline est utilisée pour interpoler les positions des sommets de manière lisse. À partir de cette technique, Funkhouser et al. [95] ont élaboré un nouveau type de modélisation géométrique guidée par les données. Partant d'un maillage initial, l'utilisateur peut sélectionner les parties qu'il souhaite éditer grâce à un algorithme de segmentation interactif, et interroger une base de données pour obtenir des objets comportant des parties similaires. Les parties souhaitées peuvent alors être extraites des modèles retrouvés et fusionnées avec le maillage de base.

Métamorphose Le mélange de modèles polygonaux est aussi lié aux techniques d'interpolation de formes et de métamorphose. Un maillage source et un maillage cible peuvent être localement interpolés pour obtenir des effets de mélange local. Les travaux dans ce domaine incluent les travaux d'Alexa [8], dans lequel les coordonnées laplaciennes sont interpolées linéairement. Xu et al. [196] procèdent par interpolation non-linéaire de champs de gradient en résolvant des équations de Poisson définies sur les maillages. Cette approche implique de nombreux problèmes de paramétrisation et de remaillage [6]. La

principale limite du mélange de maillage par les méthodes de métamorphose est que les maillages source et cible doivent avoir la même topologie.

1.2.4 Représentations différentielles

Récemment, de nouvelles méthodes de déformation et de fusion de surfaces maillées s'appuyant sur des représentations différentielles ont été développées. Celles-ci trouvent généralement leurs fondements dans la théorie spectrale des graphes. Les coordonnées différentielles telles que les coordonnées *laplaciennes* utilisées par Sorkine et al. [179] codent la position \mathbf{p}_i de chaque sommet v_i du maillage relativement à son voisinage :

$$\delta \mathbf{p}_i = \sum_{v_j \in N(v_i)} w_{i,j} (\mathbf{p}_j - \mathbf{p}_i)$$

où $\delta \mathbf{p}_i$ est l'opérateur laplacien discret au point \mathbf{p}_i , $N(v_i)$ est l'ensemble des sommets du 1-voisinage de v_i , et les coefficients $w_{i,j}$ sont des poids associés à chaque \mathbf{p}_j tels que $\sum_{v_j \in N(v_i)} w_{i,j} = 1$. Ces coordonnées ont la propriété d'encoder la fréquence du signal géométrique. Lorsqu'une déformation de type FFD est appliquée à un modèle, les détails de la géométrie peuvent être maintenus en minimisant une mesure d'erreur entre les coordonnées laplaciennes du modèle source et du modèle transformé. Le même principe peut être appliqué entre deux modèles que l'on souhaite fusionner. Il assure alors une transition lisse entre la géométrie des deux modèles. Une autre formulation par manipulation de champs de gradient et résolution de l'équation de Poisson a été proposée par Yu et al. [197]. Dans ces méthodes, le mélange de maillages nécessite de résoudre des problèmes de mise en correspondance de sommets entre les bords correspondants. Les maillages qui sont mélangés doivent avoir des tailles d'arêtes proches, ce qui nécessite en général une phase initiale de remaillage.

Méthodes multireprésentation

Ce chapitre présente un état de l'art des différentes méthodes multireprésentation impliquant des modèles implicites, des surfaces maillées et des échantillons de points. Ces deux aspects ont déjà fait l'objet de nombreuses études depuis une vingtaine d'année, avec le développement de deux types de méthodes : des méthodes de conversion entre modèles et des méthodes hybrides.

Les techniques de conversion depuis un modèle vers un autre permettent à des objets dans différentes représentations de coexister et d'interagir dans le même environnement à travers une représentation unique. Les développements récents en modélisation à partir de surfaces implicites, de maillages et de modèles représentés par des points ont conduit à de nouvelles solutions permettant de rapprocher ces différentes représentations. Des méthodes hybrides exploitant les capacités de représentations implicites et explicites au sein d'un même modèle ont également été développées.

2.1 Techniques de conversion

Nous nous intéresserons dans cette section aux techniques de conversion de maillages en surfaces implicites et à celles de surfaces implicites en maillage. Les techniques de reconstruction de surface maillées et de surfaces implicites à partir d'échantillons de points ont été présentées dans le chapitre 1 de la première partie de la thèse, aussi, nous ne les redévelopperons pas ici. La génération d'échantillons de points sur une surface, communément appelée échantillonnage de surface, est un thème de recherche important de ces dernières années, connecté au maillage et au remaillage de surface, ainsi qu'à la reconstruction de surface. Il en sera ici question dans la section consacrée au maillage de surfaces implicites. Nos travaux n'étant pas directement concernés par le problème de la génération d'échantillons de points à partir de surfaces maillées, nous renvoyons le lecteur à l'état de l'art sur le remaillage récemment proposé par Alliez et al. [11].

2.1.1 Conversion de maillages en surfaces implicites

Distance à un maillage Le problème de la conversion d'un maillage polygonal en surface implicite peut être envisagé en considérant simplement la distance euclidienne à sa frontière pour générer un champ de distances. La distance euclidienne à un maillage est en général calculée analytiquement comme le minimum des distances signées à chaque triangle. Parcourir linéairement l'ensemble des triangles

présente un coût prohibitif pour des évaluations fréquentes. Plusieurs techniques d'accélération ont ainsi été développées, principalement fondées sur des décompositions hiérarchiques de l'espace. Un état de l'art détaillé de ces techniques peut être trouvé dans [107]. Des méthodes spécifiques ont été également développées pour la construction de champs de distances échantillonnés sur une grille de l'espace. Chaque point de la grille contient la distance au point le plus proche de la surface. Des algorithmes de propagation de distances locales permettent de calculer rapidement une approximation de la distance euclidienne à la surface [137]. La construction de champ de distances directement à partir de la frontière d'un maillage est simple et robuste. L'isosurface coïncide rigoureusement avec la frontière du maillage. Cependant, la surface construite n'est pas lisse.

Reconstruction de surface implicite à partir d'un maillage Les techniques reconstruisant une surface implicite à partir d'un échantillon de points peuvent s'appliquer directement à un maillage sous des conditions d'échantillonnage suffisantes. Dans ce cas, l'information de connectivité peut être utile pour estimer des normales de manière fiable, avec une orientation consistante.

La surface linéaire par morceaux que représente un maillage peut aussi servir de base à la reconstruction de surfaces implicites avec des propriétés de continuité contrôlables. Shen et al. [176] ont ainsi développé une méthode qui construit une surface implicite à partir d'un maillage polygonal en procédant par approximation polynomiale locale selon un schéma de partition de l'unité, précédemment évoqué dans le cadre des méthodes de reconstruction à partir d'échantillons de points. Les contraintes pour l'ajustement des approximations locales sont ici définies à la fois sur les points attachés aux sommets et par intégration sur les facettes. La méthode proposée permet de contrôler le comportement interpolant ou approximant de la surface résultante ainsi que le niveau de détail de la reconstruction. Cette technique est plus robuste que la reconstruction à partir de points seulement, mais le coût global de la méthode est beaucoup plus élevé du fait de l'approximation d'intégrales sur les facettes.

2.1.2 Conversion de surfaces implicites en maillages

Les méthodes de maillage de surfaces implicites ont pour objectif de construire une approximation linéaire par morceaux de la surface. Cette représentation est nécessaire pour visualiser une surface implicite en temps interactif, et est utile pour les applications de rendu et pour la reconstruction de surfaces à partir de données échantillonnées en imagerie médicale ou en modélisation moléculaire.

La diversité des applications a conduit à développer diverses méthodes de triangulation. Ces méthodes ont en commun deux types d'opérations : une opération d'échantillonnage de la fonction potentiel, de nature géométrique, et une opération de structuration, de nature topologique. La manière dont elles sont effectuées différencie les méthodes. Nous proposons une classification en trois catégories principales : les méthodes par échantillonnage de l'espace, les méthodes par progression sur la surface, et les méthodes hybrides. Nous traitons indépendamment les travaux récents menés en géométrie algorithmique sur des techniques offrant des garanties d'approximation.

La qualité du maillage généré et le coût de la méthode sont au centre de la problématique. La qualité de la triangulation s'évalue en fonction de critères topologiques et géométriques. Idéalement, un maillage doit être topologiquement équivalent à la surface, prendre en compte les détails de sa forme, comme les arêtes vives, et générer des triangles non aplatis. Des schémas d'échantillonnage et des structures combinatoires adaptées sont nécessaires pour localiser la surface et construire des relations de connectivité, pouvant bénéficier de la connaissance de propriétés de la fonction potentiel ou de la géométrie locale.

Méthodes par échantillonnage de l'espace Les méthodes par échantillonnage de l'espace sont les plus populaires, notamment avec la méthode des Marching Cubes proposée en 1987 par Lorensen et Cline [133], dont l'idée est d'abord apparue en 1986 dans un article de Wyvill et al. [194]. L'espace est subdivisé de manière régulière ou adaptative en cellules (cubes ou tétraèdres). Les cellules traversées par la surface sont identifiées en examinant le signe de la fonction, et triangulées individuellement. Cette méthode s'applique à des surfaces lisses. Elle est rapide, mais ne s'adapte pas à la géométrie locale, et génère de nombreux triangles aplatis. L'algorithme a été étendu à plusieurs reprises pour fournir des approximations plus précises géométriquement, mais aussi topologiquement du fait de l'existence de configurations ambiguës [149, 56, 185, 58, 180, 128]. Ces dernières années, plusieurs extensions de l'algorithme original ont été développées pour mailler des surface comportant des arêtes vives, comme par exemple les méthodes de Kobbelt et al. [122], Ju et al. [117] et Ho et al. [110].

Méthodes par progression sur la surface Les méthodes par progression sur la surface propagent des primitives en suivant la surface à partir d'une primitive-germe. Contrairement aux méthodes par échantillonnage de l'espace, chaque composante connexe doit être localisée. Ces méthodes s'appliquent de manière général à des surfaces lisses.

Les méthodes à particules construisent un système de particules sur la surface, contraint selon un principe d'attraction-répulsion. La topologie peut ensuite être obtenu par un algorithme de reconstruction de surface combinatoire. La méthode la plus développée utilisant cette approche est celle proposée par Witkin et Heckbert [189]. Ces méthodes produisent des maillages de qualité du fait d'un contrôle de la distribution des particules sur la surface, mais la stabilisation du système et la nécessité d'une étape de reconstruction dans le cas où un maillage est nécessaire les rend coûteuses.

Les méthodes par prédiction/correction construisent de manière incrémentale de nouveaux sommets directement sur la surface à partir du bord du maillage courant. L'algorithme des Marching Triangles proposé par Hilton et al. [109] distribue les sommets du maillage de façon homogène en vertu d'une contrainte locale de Delaunay sur chaque triangle. Les triangles obtenus sont quasi-équilatéraux, et le coût en temps de l'algorithme est équivalent à celui des Marching Cubes. Cependant, des détails peuvent être manqués du fait d'un pas d'échantillonnage fixe. Cette approche a été étendue à plusieurs reprises, notamment avec des versions s'adaptant à la courbure de la surface proposées successivement par Karanis et Stewart [119] et Akkouche et Galin [7]. Ces méthodes permettent de produire des maillages de qualité, mais l'évaluation ou l'estimation de la courbure locale de la surface les rend coûteuses.

Méthodes hybrides Ces méthodes moins traditionnelles cherchent à adapter un maillage initial à la surface. Ces méthodes ont en général pour vocation d’être rapides. Parmi les méthodes globales, citons celle du Shrinkwrap introduite par van Overveld et Wyvill [184], s’appliquant à des surfaces à squelette. Le maillage d’une sphère est projeté sur la surface en identifiant les points critiques. Le maillage résultant est de qualité réduite. L’utilisation de surfaces de subdivision pour raffiner un maillage grossier obtenu par Marching Cubes ont également été proposées par Velho [185] et par Wyvill et al. [192]. Le maillage initial doit refléter suffisamment les détails de la surface. Les méthodes locales proposées par Desbrun et al. [70], Crespin et al. [65], et Min et al. [141] combinent les maillages de primitives générés par des méthodes ad hoc. Ces méthodes permettent difficilement de représenter des objets complexes.

Le maillage de surfaces implicites posent ainsi de multiples problèmes géométriques et topologiques qui restent encore ouverts. En pratique, il s’agit de trouver le meilleur compromis entre la qualité du maillage généré par une méthode et son coût. Pour obtenir un maillage de qualité dans les meilleurs délais, une des tendances fortes actuelles consiste à combiner une méthode de type Marching Cubes de base avec des techniques d’optimisation de maillage spécifiques proposées par Ohtake et al. [151, 153]. Cette approche s’avère actuellement plus efficace que les approches de types particules ou Marching Triangles, permettant de surcroît de traiter correctement des surfaces comportant des arêtes vives.

Maillage avec garanties d’approximation Certaines applications nécessitent des garanties sur la qualité des maillages générés relativement aux propriétés géométriques et topologiques de la surface implicite. Des travaux récents en Géométrie Algorithmique ont été consacrés à des algorithmes pouvant offrir de telles garanties. Boissonnat et Oudot [42, 43, 44] ont ainsi revisité la technique d’échantillonnage de surface de Chew [57] pour générer des triangulations optimales pour une surface implicite. La méthode proposée nécessite de pouvoir calculer ou estimer le local feature size de la surface afin de régler la densité de l’échantillonnage, ce qui est une tâche difficile pour une surface quelconque. Le même algorithme en considérant un critère sur les variations locales de la géométrie pour des surfaces k -lipschitziennes a été ensuite développé par les mêmes auteurs [45]. Cependant, les moyens à mettre en œuvre pour implanter cet algorithme n’ont pas encore été précisés. Boissonnat et al. [41] ont également développé un algorithme reposant sur la théorie de Morse garantissant que la surface implicite et son approximation sont isotopes. Il s’agit d’une étude théorique pour laquelle aucune implantation n’a été réalisée.

2.2 Méthodes hybrides

Selon les modèles géométriques, certaines opérations ou traitements ne peuvent pas être accomplis facilement de manière directe. Dans certains cas, ce problème peut être résolu en utilisant une représentation intermédiaire différente. Dans le domaine des algorithmes de traitements géométriques, certains problèmes spécifiques peuvent être résolu de manière efficace en utilisant une approche de modélisation hybride combinant représentations implicites et paramétriques [121]. Un certain nombre de méthodes hybrides ont été également développées pour la création de formes complexes, avec ces dernières années

une attention particulière portée à l'édition interactive de surfaces représentées par des points à l'aide de modèles de surface implicite.

Fusion de maillages à l'aide d'un modèle de surface implicite Deux maillages peuvent être fusionnés en un nouvelle forme en construisant une représentation implicite intermédiaire définie à partir des maillages, puis en leur appliquant les opérations de mélange classiques des surfaces implicites. Decaudin [69] a proposé la première méthode de cette nature, dans laquelle les maillages en entrée sont des polyèdres étoilés par rapport à un centre. La représentation implicite de chaque objet à fusionner est définie à partir de la distance entre son centre et sa surface. La surface résultant de la fusion est retriangulée pour obtenir un nouveau maillage. Singh et Parent [178] ont proposé une technique similaire mais plus locale. Des régions de mélange de propriétés différentes peuvent être définies à partir plusieurs centres par rapports auxquels les maillages en entrée doivent être localement étoilés. Les régions de mélange sont retriangulée localement en utilisant l'algorithme des Marching Cubes.

On peut souligner que ces méthodes sont en fait indépendantes de la représentation des entrées, puisque la seule exigence est de savoir calculer l'intersection d'un rayon avec une surface. Si la représentation implicite intermédiaire utilisée est restrictive, une idée intéressante est l'utilisation des techniques standard de mélange de surfaces implicites avec une méthode locale de retriangulation. Dans notre modèle hybride, nous avons adopté des principes similaires pour la fusion de différents objets, avec également une approche locale du maillage des régions de mélange entre les objets.

Déformation de maillage à l'aide d'un modèle de surface implicite Crespin [64] a proposé le modèle des IFFD (*Implicit FFD*) pour déformer localement ou globalement des surfaces maillées. Les déformations sont définies à partir de primitives implicites à squelette placées par l'utilisateur. En mélangeant les primitives, le champ de potentiel généré est considéré comme déformant l'espace. L'objet qui y est plongé subit la déformation en conséquence à la manière des FFD.

Déformation locale de surface implicite à l'aide d'un modèle paramétrique Différentes techniques permettant de simuler des déformations libres à l'aide de champs de potentiels spécifiques ont été développées. En particulier, Schmitt et al. [172] ont proposé une méthode pour déformer localement des surfaces implicites définies d manière fonctionnelle en utilisant des primitives à squelette construites à partir de courbes comme des splines et des fonctions de distance particulières. Cette approche ne permet cependant pas un contrôle aussi intuitif que les outils de déformation de maillage et n'offrent pas autant de degrés de liberté.

Edition de surfaces représentées par des points Ces dernières années, plusieurs travaux se sont focalisés sur l'édition interactive de surfaces représentées par des échantillons de points sans usage de structure combinatoire.

Reuter et al. [162] reconstruisent une surface implicite à partir d'un échantillon de points en utilisant des RBF à support compact (première partie, ch. 1). Les formes obtenues peuvent être contrôlées localement de manière intuitive et interactive en agissant directement sur les points de contrainte. La mise en œuvre d'une structure hiérarchique permet une réévaluation locale de la fonction potentiel. Cependant, cette approche demeure coûteuse pour manipuler des échantillons de points denses représentant des formes détaillées.

Pauly et al. [161] ont développé une vaste panoplie d'outils d'édition interactive de surfaces représentées par des points utilisant le modèle de surface MLS (première partie, ch. 1), plus souple à manipuler que les RBF. Les outils comprennent des opérations booléennes et de mélange, ainsi que des déformations de type FFD. Les opérations de combinaison exploitent la représentation implicite, tandis que les déformations sont accomplies en déplaçant les points et en mettant à jour la surface et l'échantillon dynamiquement. L'intérêt majeur de cette approche est qu'elle permet de manipuler des objets numérisés sans reconstruction d'une surface globale. Aucune information sur la topologie n'est maintenue explicitement, ce qui présente des avantages pour certaines opérations conduisant à des changements topologiques. A chaque fois qu'une information relative au voisinage d'un point est nécessaire, des relations de connectivité définies à partir des k plus proches voisins sont calculées à la volée en utilisant une structure de recherche spatiale adaptée. Cependant, l'établissement d'une connectivité correcte n'est pas toujours un problème trivial, qui dépend de la densité et de la distribution des points sur la surface. En outre, la densité de l'échantillon doit être mise à jour fréquemment pour maintenir une surface cohérente, ce qui implique de nombreux appels à la procédure de projection. Rappelons aussi que ce modèle ne permet pas de représenter facilement des surfaces avec des arêtes vives.

Ces méthodes mettent en œuvre une visualisation par splatting qui nécessite de maintenir un échantillon de points suffisamment dense des surfaces manipulées, et de l'adapter au point de vue, ce qui ajoute à leur complexité pour manipuler interactivement des données de taille très importante.

2.3 Synthèse

Dans ce chapitre, nous avons présenté les principales techniques permettant d'exploiter la complémentarité des surfaces implicites et des maillages ou des échantillons de points, soit par conversion, soit en utilisant des méthodes hybrides.

Les méthodes de conversion permettent de s'affranchir des limites des modèles existants en permettant de passer d'une représentation à une autre, mais elles posent des problèmes de coût lorsqu'elles sont globales pour des objets détaillés et peuvent manquer de fiabilité à cause d'une possible perte d'information.

L'idée de faire coopérer les modèles pour bénéficier de leurs avantages respectifs dans les opérations d'édition offre davantage d'interactivité. Différentes méthodes hybrides ont été développées pour des opérations spécifiques, mais certaines sont restrictives, ou dépendantes de la représentation des entrées dans la mesure où il n'est pas forcément possible d'obtenir un résultat équivalent à partir d'une

représentation implicite ou explicite.

Le modèle de surface MLS appliqué à des échantillons de points permet de bénéficier en même temps de tous les avantages d'une représentation explicite, grâce à la manipulation directe de points sur la surface, et de ceux d'une représentation implicite. Plus qu'une méthode, il s'agit d'une représentation hybride. Elle n'est cependant pas dénuée d'inconvénients, notamment liés à l'absence de connectivité. Tous les objets doivent également être manipulés dans la même représentation sous la forme d'échantillons de points.

Dans cette thèse, notre objectif est différent. Nous voulons manipuler des surfaces implicites, des maillages et des échantillons de points en conservant ces représentations, et en appliquant des opérations en utilisant la représentation la plus adaptée. Nous proposons de mettre en œuvre des conversions locales, préservant les surfaces existantes, et permettant plus d'interactivité que des conversions globales. Nous proposons un système reposant sur un arbre de construction que nous allons décrire dans le chapitre suivant.

L'HybridTree :

un arbre de construction hybride

3.1 Architecture du modèle

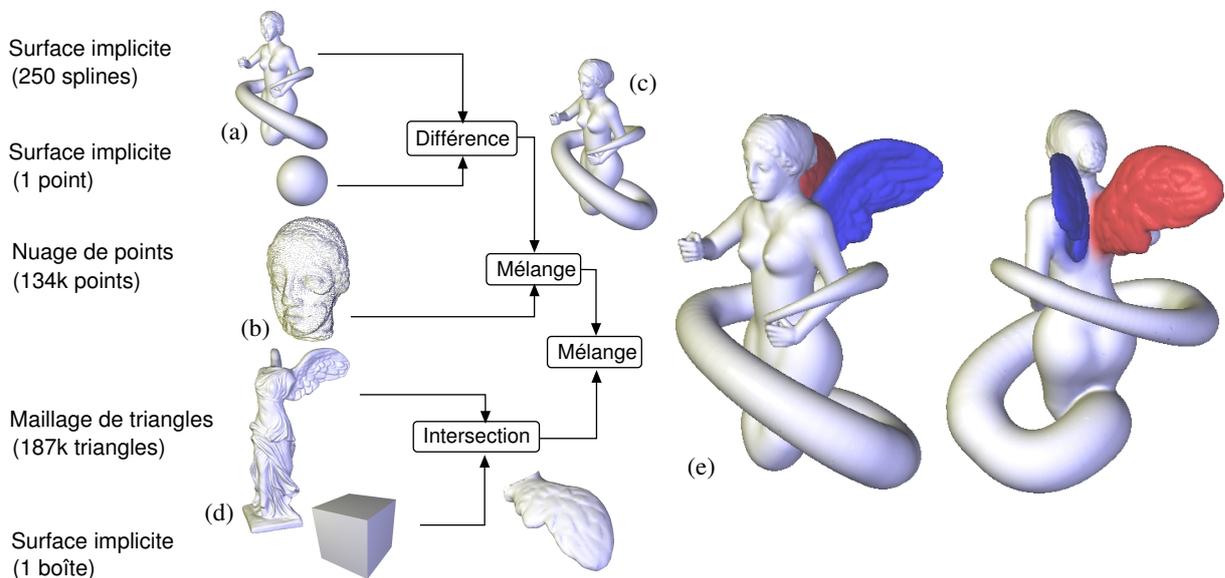


FIG. 3.1 – La structure d'HybridTree pour un modèle de Méduse ailée.

L'HybridTree est un arbre de construction hiérarchique dont les feuilles portent soit des surfaces implicites, soit des maillages polygonaux, soit des nuages de points. Ces modèles sont combinés au niveau des nœuds de l'arbre au moyen d'opérations de mélange, d'opérations booléennes, ou bien d'opérations de déformation de l'espace. Les déformations comprennent les transformations affines, les déformations de Barr et des déformations libres de type FFD. Les opérations constructives sont des nœuds binaires, tandis que les opérations de déformation sont des nœuds unaires.

La figure 3.1 montre la structure d'HybridTree pour un modèle de Méduse ailée. Le modèle de la MÉDUSE (a) a été entièrement construit à partir de primitives implicites à squelette. En utilisant une différence booléenne, seul le corps a été conservé, et a été mélangé avec le modèle IGEA (b) pour obtenir le résultat en (c). Les ailes d'un modèle maillé de la VICTOIRE DE SAMOTHRACE (a) ont été extraites par intersection avec une boîte. Les ailes et le modèle de MÉDUSE modifié ont été finalement mélangés

pour obtenir le modèle en (e).

L'HybridTree est évalué de manière incrémentale en parcourant la structure d'arbre récursivement. L'architecture de notre système d'évaluation est représentée dans la figure 3.2. Chaque pôle correspond à une représentation géométrique qui fournit un ensemble d'opérations pour lesquelles elle est la plus adaptée. Les flèches représentent les passerelles entre les modèles, qui correspondent à des procédures de conversion. En partant d'un objet implicite, maillé ou sous la forme d'un nuage de points, celui-ci est d'abord converti à la volée dans la bonne représentation avant d'appliquer une opération. Les passerelles disponibles dans notre implantation actuelle sont représentées par les flèches solides dans le diagramme. Les surfaces implicites et les maillages constituent le cœur de notre système. Tous les nœuds de l'HybridTree peuvent générer à la fois un champ de potentiel dans l'espace et un maillage. Les nuages de points sont intégrés par l'intermédiaire d'une méthode de reconstruction de surface qui produit un maillage. Notre système peut ainsi être rendu complet par transitivité.

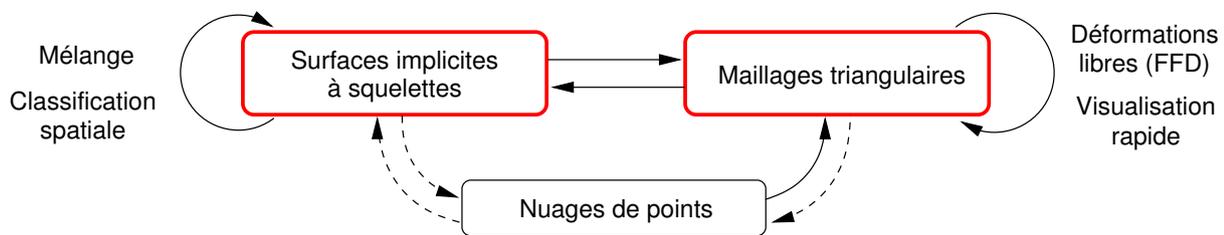


FIG. 3.2 – Le système d'évaluation de l'HybridTree.

Les modèles représentés par des échantillons de points sont convertis en utilisant la technique de reconstruction sélective présentée dans la première partie de cette thèse, qui offre un contrôle sur la précision du maillage résultant. Une application de cette technique est illustrée dans la figure 3.3. L'échantillon de point en entrée comporte 134 344 points, presque uniformément distribués sur la surface. Les images au centre et à droite montrent le résultat de la reconstruction sélective. Un niveau de détail ayant été fixé, l'échantillon de points a été automatiquement simplifié au cours du processus de reconstruction de surface. Le modèle reconstruit a 26 823 sommets, soit une réduction de 76% par rapport au nombre de points du modèle de départ, et 53 728 triangles. Le résultat en été obtenu en 15 secondes sur une machine équipée d'un processeur Pentium IV 3.0 GHz avec 1 Go de RAM.

L'HybridTree est évalué à travers trois requêtes fondamentales et un processus de maillage implantés de manière spécifique pour chaque type de nœud. Les requêtes d'évaluation de la fonction potentiel en un point de l'espace sont accomplies à chaque fois que la formulation implicite est nécessaire. Nous avons recours à cette formulation essentiellement pour les opérations de mélange. Les requêtes de gradient permettent d'obtenir la normale exacte en des points d'échantillonnage. Certaines opérations nécessitent seulement de savoir si un point est à l'intérieur ou à l'extérieur de la surface, comme les opérations booléennes. Au lieu d'évaluer le signe de la fonction de distance explicitement, nous avons développé des méthodes spécifiques pour accélérer l'évaluation de ce type de requête. Le dernier type de requête dans notre système est un processus de maillage incrémental invoqué en un nœud donné si une représentation explicite de la surface est nécessaire, pour les déformations locales ou la visualisation interactive. Les

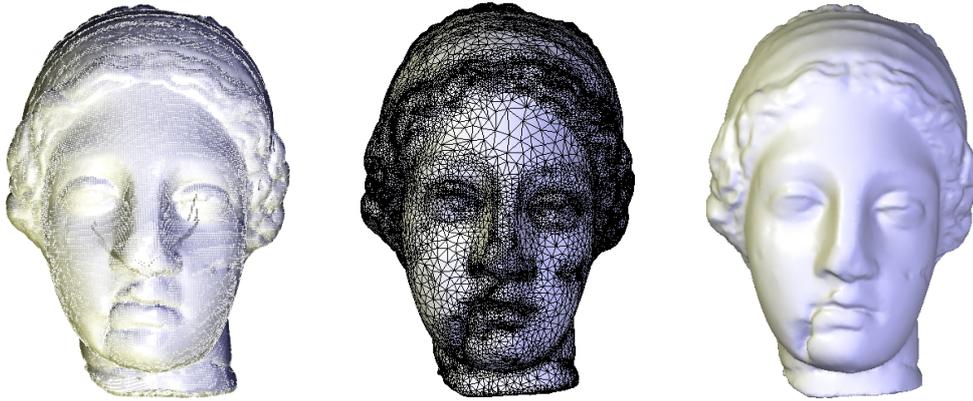


FIG. 3.3 – Illustration de notre technique de reconstruction sélective sur le modèle IGEA. À gauche : échantillon de points original (134 344 points); Au centre et à droite : maillage reconstruit (26 823 sommets, 53 728 triangles).

résultats locaux sont combinés de manière cohérente par les différentes opérations constructives.

Le reste de ce chapitre est organisé comme suit. Dans la section 3.2, nous présentons nos algorithmes d'évaluation pour les requêtes fondamentales, puis nous détaillons nos algorithmes de maillage dans la section 3.3. Nous présentons enfin plusieurs applications de l'HybridTree pour la modélisation de formes complexes dans la section 3.5 avant de conclure dans le chapitre 3.

Notations La fonction potentiel associée à un nœud ou à un modèle A sera notée f_A , et le gradient correspondant ∇f_A . Nous appellerons $c_A(\mathbf{p})$ la fonction de classification spatiale d'un point \mathbf{p} par rapport à la surface du modèle A (*Point Membership Classification*), qui peut prendre trois valeurs différentes $\{1, 0, -1\}$ selon que \mathbf{p} se trouve respectivement à l'intérieur, sur, ou à l'extérieur de la surface de A . Le maillage de la surface de A sera noté \mathcal{M}_A , et la boîte englobante du modèle A sera notée \mathcal{B}_A .

3.2 Requêtes fondamentales

Dans les paragraphes suivants, nous expliquons comment les fonctions potentiel, gradient et de classification spatiale sont évaluées pour les différents types de nœuds dans l'HybridTree.

3.2.1 Primitives implicites à squelette

Les primitives implicites que nous manipulons sont des primitives à squelette. La fonction potentiel pour un squelette donné est définie de manière analytique en utilisant la formulation des surfaces implicites à squelette, c'est-à-dire :

$$f(\mathbf{p}) = g \circ d(\mathbf{p})$$

où $d : \mathbb{R}^3 \rightarrow \mathbb{R}_+$ représente la distance euclidienne au squelette, et $g : \mathbb{R}_+ \rightarrow \mathbb{R}$ la fonction de distribution du potentiel dans l'espace. Dans notre système, nous utilisons des fonctions de distribution du potentiel polynomiales à support compact de la forme :

$$g(r) = \begin{cases} I \left(1 - \frac{r^2}{R^2}\right)^n, & n \geq 2 \quad \text{si } r \in [0, R] \\ 0 & \text{sinon} \end{cases}$$

Ces fonctions de distribution sont paramétrisées par une valeur de potentiel maximale $I \in \mathbb{R}$ atteinte en tout point du squelette et par un rayon d'influence $R \in \mathbb{R}_+$. La région d'influence associée à un squelette, caractérisée par un potentiel non nul, est notée Ω . Les fonctions inverses correspondantes $g^{-1} : \mathbb{R} \rightarrow \mathbb{R}_+$ sont définies comme suit :

$$g^{-1}(t) = \begin{cases} R \sqrt{1 - \left(\frac{t}{I}\right)^{\frac{1}{n}}}, & n \geq 2 \quad \text{si } 0 < t \leq I \text{ ou } I \leq t < 0 \\ 0 & \text{sinon} \end{cases}$$

Les normales peuvent être obtenues directement à partir du gradient de la fonction potentiel ∇f qui est évaluée comme suit :

$$\nabla f(\mathbf{p}) = g' \circ d(\mathbf{p}) \nabla d(\mathbf{p})$$

Comme la fonction d est la distance euclidienne, $\nabla d(\mathbf{p})$ est calculé comme le vecteur entre la projection orthogonale de \mathbf{p} sur le squelette et \mathbf{p} .

L'HybridTree implante un grand nombre de squelettes complexes, incluant des courbes, des surfaces et des volumes hérités du BlobTree [27]. La figure 3.4 montre une bouteille construite en utilisant des squelettes de type surface de révolution, spline, cercle et cylindre creux. Tous les types de primitives implantent un algorithme spécifique qui calcule analytiquement la distance $d(\mathbf{p})$ à son squelette de manière optimisée. Bien que le calcul de la distance entre un point de l'espace et un squelette simple comme un point ou un segment de droite soit direct et efficace, les algorithmes deviennent plus sophistiqués à mesure que la complexité des squelettes augmente. Des algorithmes détaillés peuvent être trouvés dans l'ouvrage de Schneider et Eberly [173] et dans l'article de Barbier et Galin [26].

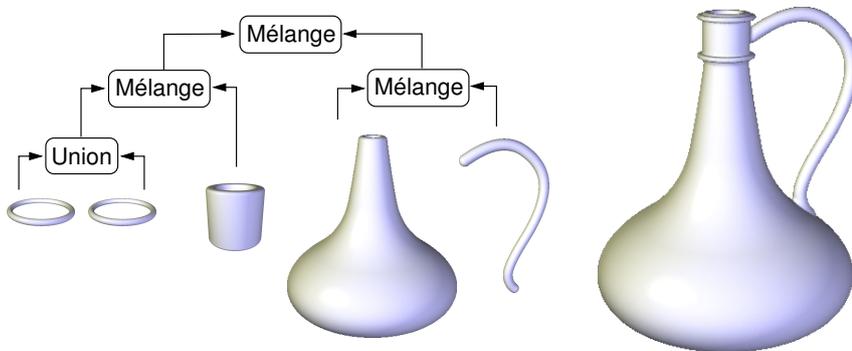


FIG. 3.4 – Modèle de bouteille construit en utilisant des primitives à squelette complexe.

Pour les surfaces implicites à squelette, la classification spatiale d'un point par rapport à la surface est généralement obtenue par l'intermédiaire de l'évaluation de la fonction potentiel, en comparant sa valeur à celle du seuil. Comme nous manipulons des primitives à squelette, nous n'avons pas besoin d'évaluer la fonction potentiel complète pour les requêtes de classification spatiale. La surface niveau T de ces primitives peut en effet être définie en balayant une sphère de rayon constant $r_T = g^{-1}(T)$ le long de la frontière du squelette. Cette isosurface existe si et seulement si $r_T \geq 0$. Dès lors, pour un point \mathbf{p} dans l'espace, la fonction de classification spatiale c peut être définie comme suit :

$$c(\mathbf{p}) = \begin{cases} -1 & \text{si } 0 \leq r_T < d(\mathbf{p}) \\ 1 & \text{si } 0 < d(\mathbf{p}) < r_T \\ 0 & \text{sinon} \end{cases}$$

Le rayon r_T est calculé une seule fois pour chaque primitive. Pour les primitives à squelette volumique, la position des points de requête par rapport à l'intérieur du squelette est obtenue de manière analytique au cours du calcul de la distance $d(\mathbf{p})$.

Les primitives implicites à squelette sont définies uniquement par un petit nombre de paramètres, ce qui permet un coût de stockage moindre. Les squelettes complexes sont aussi plus facile à contrôler qu'une combinaison de primitives simples, mais l'évaluation des fonctions de distance devient plus coûteuse. Une technique d'accélération simple consiste à précalculer et stocker des résultats nécessaires à plusieurs évaluations, par exemple le long d'un rayon [27]. Quand l'évaluation de f et ∇f sont toutes deux requises, des résultats intermédiaires communs sont également calculés une seule fois. À l'échelle de l'arbre de construction, chaque primitive est équipée d'une boîte englobante qui permet aussi d'éviter des évaluations inutiles de la fonction potentiel.

3.2.2 Maillages polygonaux

Pour les maillages, la fonction potentiel est évaluée en utilisant la même formulation que pour les primitives implicites à squelette. La fonction de distance $d_{\mathcal{M}}$ d'un point $\mathbf{p} \in \mathbb{R}^3$ à un maillage triangulaire \mathcal{M} est définie comme la distance euclidienne minimale entre \mathbf{p} et tout triangle \mathcal{T} de \mathcal{M} :

$$d_{\mathcal{M}}(\mathbf{p}) = \min_{\mathcal{T} \in \mathcal{M}} d(\mathbf{p}, \mathcal{T})$$

La surface implicite générée par le maillage en tant que squelette pour un seuil fixé T est une surface qui diffère du maillage original \mathcal{M} . Cette surface peut être définie par le balayage d'une sphère de rayon constant $r_T = g^{-1}(T)$ le long de la frontière de \mathcal{M} . Afin de faire correspondre la frontière de \mathcal{M} avec l'isosurface de niveau T indépendamment des paramètres de la fonction de distribution du potentiel dans l'espace, nous opérons un décalage dans la fonction de distance en y incorporant le seuil, ce qui donne

lieu à la fonction de distance suivante :

$$d(\mathbf{p}) = \begin{cases} d_{\mathcal{M}}(\mathbf{p}) + r_T & \text{si } \mathbf{p} \text{ est à l'extérieur de } \mathcal{M} \\ r_T - d_{\mathcal{M}}(\mathbf{p}) & \text{si } \mathbf{p} \text{ est à l'intérieur de } \mathcal{M} \text{ et } d_{\mathcal{M}}(\mathbf{p}) < r_T \\ 0 & \text{sinon} \end{cases}$$

Cette fonction de distance garantit que l'isosurface et la frontière du maillage coïncident pour n'importe quelle valeur de T . Le mécanisme de décalage est illustré par les figures 3.5(droite) et 3.6(droite).

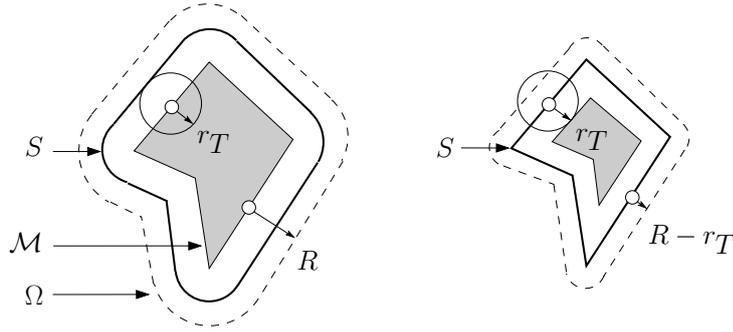


FIG. 3.5 – Mécanisme de décalage de distance sur un polygone 2D. Sur la gauche, seule la fonction de distance de base $d_{\mathcal{M}}$ est utilisée. L'effet de la fonction de distance avec décalage est illustré sur la droite.

Cette approche permet à l'utilisateur de garder le contrôle sur tous les paramètres de la fonction potentiel sans restriction. Le rayon d'influence du maillage, qui passe de R à $R - r_T$ est redimensionné de manière à rendre transparent le décalage dans la fonction de distance. Si l'utilisateur spécifie un rayon d'influence R , la fonction potentiel est évaluée avec un rayon d'influence R' plus grand tel que $R' - r'_T = R$, où $r'_T = g^{-1}(T)$ dépend de R' . Pour les fonctions de distribution du potentiel que nous manipulons, nous avons :

$$\frac{r'_T}{R'} = \frac{r_T}{R} = \sqrt{1 - \left(\frac{T}{I}\right)^{\frac{1}{n}}}$$

qui dépend uniquement de paramètres fixés. Le rayon R' est alors calculé comme suit :

$$R' = \frac{R}{\left(1 - \frac{r_T}{R}\right)}$$

Le gradient de la fonction potentiel est calculé à partir du gradient ∇d de la fonction de distance d défini comme suit :

$$\nabla d(\mathbf{p}) = \begin{cases} \nabla d_{\mathcal{M}}(\mathbf{p}) & \text{si } \mathbf{p} \text{ est à l'extérieur de } \mathcal{M} \\ -\nabla d_{\mathcal{M}}(\mathbf{p}) & \text{si } \mathbf{p} \text{ est à l'intérieur de } \mathcal{M} \text{ et } d_{\mathcal{M}}(\mathbf{p}) < r_T \\ 0 & \text{sinon} \end{cases}$$

Comme pour les primitives implicites, $\nabla d(\mathbf{p})$ est le vecteur entre la projection orthogonale de \mathbf{p} sur le maillage \mathcal{M} et \mathbf{p} , qui est obtenue en calculant $d(\mathbf{p})$.

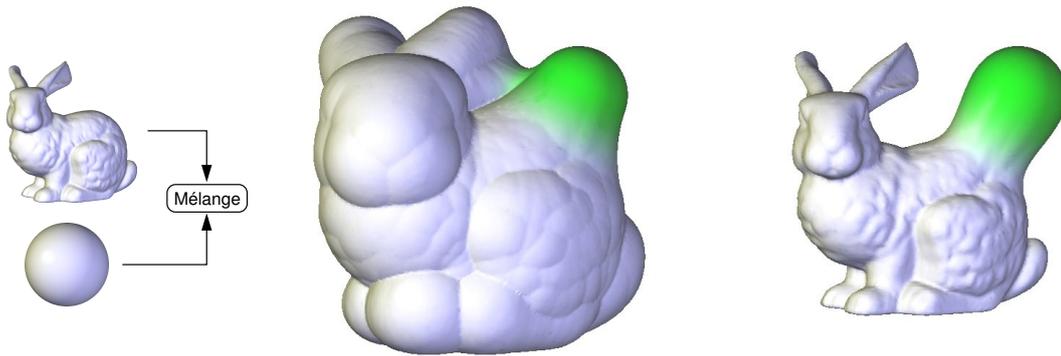


FIG. 3.6 – Le modèle maillé BUNNY (69 674 triangles) mélangé avec une sphère implicite. Le résultat à gauche est obtenu avec la distance $d_{\mathcal{M}}(\mathbf{p})$ tandis que le modèle à droite a été calculé en utilisant la pseudo fonction de distance $d(\mathbf{p})$.

Calculer la distance minimale entre un point \mathbf{p} et tous les triangles \mathcal{T} du maillage \mathcal{M} est une opération coûteuse en temps. Nous avons implanté un algorithme inspiré de la méthode de Johnson et Cohen [116]. Nous nous appuyons sur une hiérarchie de boîtes englobantes construite par partition binaire de l'espace. Cette hiérarchie est parcourue en largeur pour chaque requête de distance entre un point et le maillage. Pour chaque boîte traversée, l'algorithme calcule une borne inférieure et une borne supérieure de la distance minimale entre un point et le maillage, ce qui permet d'éliminer rapidement les régions de l'espace qui ne contiennent pas les parties du maillage les plus proches du point de requête. En outre, nous utilisons le fait que la fonction de distribution du potentiel devient nulle au delà de la distance R à la frontière du maillage de manière à réduire encore davantage le nombre de calculs de distance inutiles.

Comme illustré par la figure 3.7, l'efficacité de notre optimisation liée au rayon d'influence varie en fonction de la géométrie du maillage \mathcal{M} et la valeur de R . Nous avons mesuré le temps nécessaire pour effectuer 100^3 requêtes de distance pour plusieurs maillages, avec un rayon d'influence croissant. Pour chaque maillage, les coordonnées des sommets ont été normalisées de manière à ce que la plus grande dimension de sa plus petite boîte englobante soit unitaire. Les points de requête correspondent à un échantillonnage régulier du volume de cette boîte. Les temps correspondant aux courbes solides prennent notre optimisation en compte, alors que ce n'est pas le cas pour les temps correspondant aux courbes en pointillé. On peut observer que le temps de calcul est sensiblement réduit lorsque R est petit par rapport à la taille globale du maillage. Seul un petit nombre de boîtes de la hiérarchie nécessitent d'être testées dans ce cas. Le bénéfice de notre accélération décroît ensuite à mesure que R augmente, les temps de calcul tendant à se stabiliser au niveau des temps sans optimisation.

Ces temps de calcul ne prennent pas en compte le temps nécessaire à la construction de la hiérarchie de boîtes englobantes. Les temps de précalcul sont reportés dans la table 3.1, qui indique aussi la profondeur de la hiérarchie de boîtes englobantes pour les différents modèles que nous avons testés. Les boîtes situées au niveau des feuilles contiennent au plus 4 triangles de manière à éviter tout surcoût lié aux tests sur les boîtes englobantes.

Le coût des calculs de distance entre un point et un maillage pourrait être encore davantage réduit

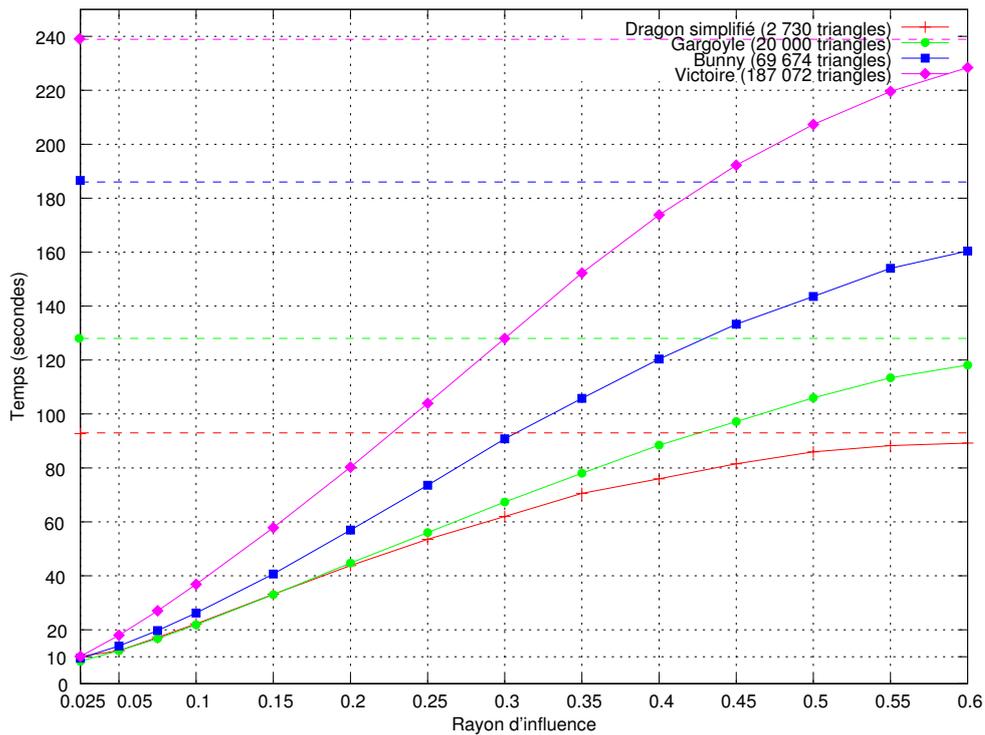
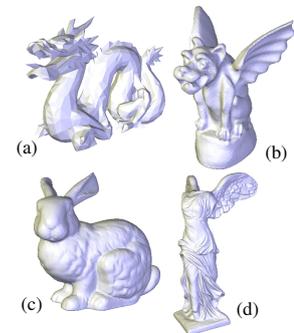


FIG. 3.7 – Temps de calcul pour 100^3 requêtes de distance entre un point de l'espace et un maillage (calculs effectués sur un processeur Pentium IV 3.0 GHz avec 1 Go de RAM).

Modèle	Nb. triangles	Profondeur	Temps prétraitement
DRAGON simplifié (a)	2,730	23	0.05
GARGOYLE (b)	20,000	22	0.39
BUNNY (c)	69,674	28	1.31
VICTOIRE (d)	187,072	25	4.82



TAB. 3.1 – Nombre de triangles, profondeur de la hiérarchie de boîtes englobantes et temps de prétraitement (en secondes) pour quatre maillages différents.

en utilisant une décomposition du maillage en *éventails* [96], qui permet d'économiser des opérations en factorisant les calculs entre les triangles voisins dans chaque éventail de triangles.

La classification spatiale est obtenue en calculant le nombre d'intersections entre un rayon et le maillage en utilisant la même hiérarchie de boîtes englobantes. Pour réduire le nombre de cellules à traverser, la direction et l'orientation du rayon sont choisies de manière à ce que la distance entre le point de requête et le point d'intersection entre le rayon et la boîte englobante du maillage soit minimale.

3.2.3 Opérations de mélange

Dans notre système, nous proposons deux types de formulations pour mélanger deux surfaces : une globale et une locale. Soient A et B deux modèles à mélanger ensemble. Le mélange global entre les deux objets est défini comme proposé à l'origine par Blinn [36] :

$$f_{A+B} = f_A + f_B$$

La figure 3.8 montre deux modèles maillés mélangés en utilisant l'opération de mélange global. Le principal inconvénient de cette approche de base est le problème bien connu du mélange indésirable [49]. Comme montré dans l'image de droite dans la figure 3.8, un raccord non souhaité s'est créé entre l'aile droite et l'oreille droite du BUNNY.

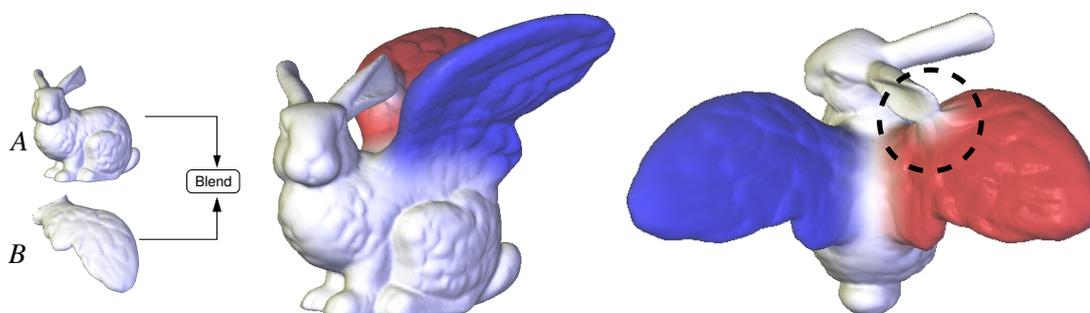


FIG. 3.8 – Mélange global entre deux modèles maillés : le BUNNY (69 674 triangles) et une paire d'ailes (8 170 triangles).

Pour effectuer le mélange avec un meilleur contrôle local, nous avons implanté une nouvelle opération de mélange local adaptée de la technique décrite par Pasko et al. dans [158]. Ce nœud de mélange a trois fils. Les deux premiers, notés A et B , représentent les deux modèles qui vont partiellement se mélanger, tandis que le troisième, noté C , représente la région de l'espace dans laquelle le mélange se réalise effectivement.

Dans notre système, C est caractérisé par un champ de potentiel défini comme l'union de primitives implicites C_i . Une telle combinaison permet de construire des régions de mélange complexes avec un résultat prédictible. La fonction potentiel f_C caractérise la région de mélange et est utilisée pour contrôler le raccord entre les deux sous-arbres correspondant à A et B . Les valeurs prises par la fonction potentiel f_{C_i} doivent être comprises entre 0 et 1. En un point \mathbf{p} de l'espace, si $f_C(\mathbf{p}) = 0$, alors le résultat est localement celui d'une union entre A et B , ce qui se produit pour tout point en dehors de la région d'influence de C . Si $f_C(\mathbf{p}) = 1$, alors le mélange se produit.

L'évaluation de cette opération de mélange local est réalisé comme suit. Tout d'abord, nous calculons la valeur du potentiel qui résulte du mélange global entre A et B , i.e. $f_{A+B}(\mathbf{p}) = f_A(\mathbf{p}) + f_B(\mathbf{p})$, ainsi que celle qui résulte de leur union, i.e. $f_{A \cup B}(\mathbf{p})$. Nous définissons la fonction potentiel résultante comme

une moyenne pondérée :

$$f_{A+B}(\mathbf{p}) = f_C(\mathbf{p}) f_{A+B}(\mathbf{p}) + (1 - f_C(\mathbf{p})) f_{A \cup B}(\mathbf{p})$$

Les primitives construites à partir d'un squelette volumique sont particulièrement utiles pour définir des régions de mélange local. Dans la figure 3.9, la région de mélange local C est définie comme l'union de deux cylindres implicites.

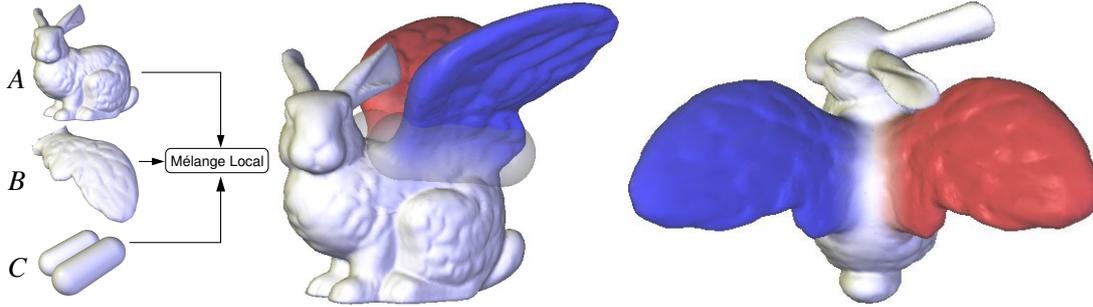


FIG. 3.9 – Mélange local. Deux cylindres implicites définissent la région de mélange local entre le BUNNY et la paire d'ailes.

Pour les deux techniques de mélange, le gradient est obtenu en dérivant les fonctions potentiel. Pour la classification spatiale, nous distinguons les champs de potentiel *positifs* et *négatifs*. On dira qu'un modèle A génère un champ de potentiel positif si l'une de ses primitives est telle que $I > 0$. On dira au contraire qu'un modèle A génère un champ de potentiel négatif si et seulement s'il ne consiste que de primitives telles que $I < 0$.

Soient A et B deux modèles qui se mélangent globalement tels que A et B génèrent tous deux des champs de potentiel positifs. Alors la classification spatiale est réalisée de la manière suivante :

1. Si \mathbf{p} se trouve à l'intérieur de $\mathcal{B}_A \cap \mathcal{B}_B$, alors calculer $v = f_{A+B}(\mathbf{p})$.
 - (a) Si $v < T$, alors \mathbf{p} se trouve à l'intérieur de la surface : retourner 1.
 - (b) Sinon, si $v > T$, \mathbf{p} se trouve à l'extérieur de la surface : retourner -1 .
 - (c) Sinon, \mathbf{p} se trouve sur la surface : retourner 0.
2. Sinon, si \mathbf{p} appartient à $\mathcal{B}_A \setminus \mathcal{B}_B$ (respectivement $\mathcal{B}_B \setminus \mathcal{B}_A$), alors effectuer la requête de classification spatiale sur A (respectivement sur B) et retourner le résultat.
3. Sinon, \mathbf{p} se trouve à l'extérieur de la surface : retourner -1 .

Pour les nœuds de mélange local tels que A , B et C génèrent des champs de potentiel positifs, l'algorithme précédent est légèrement modifié. Dans l'étape 1, nous considérons la boîte englobante \mathcal{B}_C de la région de mélange et nous évaluons f_{A+B} . Dans l'étape 2, nous considérons les boîtes $\mathcal{B}_A \setminus \mathcal{B}_C$ et $\mathcal{B}_B \setminus \mathcal{B}_C$.

Pour les nœuds de mélange global tels que A génère un champ de potentiel positif et B génère un champ de potentiel négatif, l'algorithme précédent est modifié comme suit : dans l'étape 2, si \mathbf{p} appartient

à $\mathcal{B}_B \setminus \mathcal{B}_A$, alors \mathbf{p} est à l'extérieur de la surface. Si \mathbf{p} appartient à $\mathcal{B}_A \setminus \mathcal{B}_B$, alors la requête de classification spatiale est effectuée sur A . Si A et B génèrent tous deux des champs de potentiel négatifs, alors \mathbf{p} se trouve à l'extérieur de la surface.

3.2.4 Opérations booléennes

Les fonctions potentiel proposées par Ricci [163] constituent le moyen le plus simple et le plus efficace d'effectuer des opérations booléennes entre des surfaces implicites :

$$\begin{aligned} f_{A \cup B} &= \max(f_A, f_B) \\ f_{A \cap B} &= \min(f_A, f_B) \\ f_{A \setminus B} &= \min(f_A, -f_B) \end{aligned}$$

Les fonctions min et max présentes dans ces fonctions potentiel introduisent cependant des discontinuités dans le champ de potentiel. Outre le fait que le gradient n'est pas défini au niveau de ces discontinuités, celles-ci se propagent de manière indésirable aux régions de mélange (fig. 3.10, à gauche).

Au contraire des fonctions min et max, les R-fonctions définissent un champ de potentiel avec continuité C^n partout dans l'espace, excepté sur la surface, ce qui évite des discontinuités indésirables. Les fonctions prescrites dans [157] opèrent sur des fonctions potentiel avec support infini, tandis que notre modèle manipule des fonctions potentiel à support compact. Les R-fonctions ont de plus été élaborées pour des surfaces implicites caractérisées par une valeur de seuil nulle : $T = 0$.

Nous avons adapté ces fonctions à notre modèle en incorporant la valeur de seuil dans les équations de Pasko et al. [157]. Un coefficient est destiné à garantir que la fonction potentiel résultante est toujours à support compact. Nous avons :

$$\begin{aligned} f_{A \cup B} &= T + \frac{1}{2 - \sqrt{2}} \left[(f_A - T) + (f_B - T) + \sqrt{(f_A - T)^2 + (f_B - T)^2} \right] \\ f_{A \cap B} &= T + \frac{1}{2 + \sqrt{2}} \left[(f_A - T) + (f_B - T) - \sqrt{(f_A - T)^2 + (f_B - T)^2} \right] \end{aligned}$$

Bien que les fonctions min et max d'une part et les R-fonctions d'autre part génèrent des champs de potentiel différents dans l'espace, les deux représentations produisent la même surface implicite si les opérations booléennes se trouvent au sommet de la structure d'arbre. Dans ce cas, le calcul des fonctions avec min et max est moins coûteux que celui des R-fonctions. Nous utilisons les R-fonctions modifiées pour créer un champ de potentiel continûment différentiable si les nœuds de mélange sont situés au-dessus des opérations booléennes dans l'HybridTree. Notre système adapte automatiquement les fonctions utilisées pour évaluer les opérations booléennes selon le contexte durant l'évaluation. La figure 3.10 illustre les deux opérations d'union.

Pour les opérations booléennes définies en utilisant les fonctions min et max, le gradient en un point

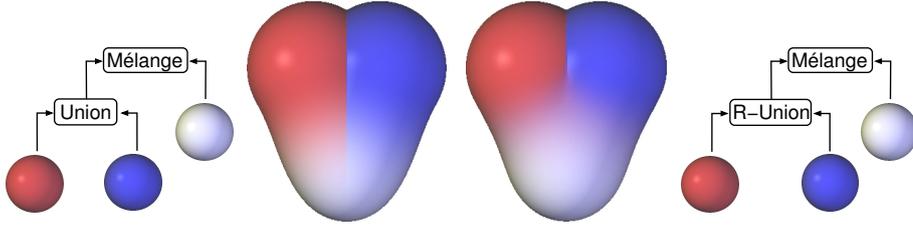


FIG. 3.10 – Avec l’opération d’union standard (à gauche), la discontinuité dans le champ de potentiel produite par la fonction max est propagée à la région de mélange. Le problème est réglé en utilisant la formulation continue par R-fonction (à droite).

de l’espace est le gradient du champ dont la contribution est maximale ou minimale entre les deux nœuds fils. En utilisant la formation par R-fonction, le gradient est simplement dérivé des fonctions potentiel.

La classification spatiale est obtenue par simple algèbre booléenne. Par exemple, la classification pour l’opération de différence entre deux modèles A et B est effectuée comme suit :

1. Si $c_A(\mathbf{p}) = 1$, alors :
 - (a) Si $c_B(\mathbf{p}) = -1$, alors retourner 1.
 - (b) Sinon, si $c_B(\mathbf{p}) = 0$, alors retourner 0.
 - (c) Sinon, retourner -1 .
2. Sinon, si $c_A(\mathbf{p}) = 0$ et $c_B(\mathbf{p}) \in \{0, 1\}$ alors retourner 0.
3. Sinon, retourner -1 .

3.2.5 Opérations de déformation

Dans notre système, une surface peut être déformée en déformant localement l’espace. Nos opérations de déformation comprennent les transformations affines, et les déformations de Barr (torsion, écrasement, courbure) [29]. Notre système offre aussi la possibilité d’appliquer des déformations locales libres de type FFD [174]. Dans les paragraphes qui suivent, A représentera l’objet associé au nœud fils d’un nœud de déformation, w une transformation de l’espace de \mathbb{R}^3 dans \mathbb{R}^3 , et w^{-1} la transformation inverse correspondante.

3.2.5.1 Déformations de Barr

Quand la formulation implicite est nécessaire, les déformations de Barr sont appliquées comme des déformations de l’espace. Le champ de potentiel résultant est défini en utilisant la fonction de déformation inverse comme suit :

$$f_w(\mathbf{p}) = f_A \circ w^{-1}(\mathbf{p})$$

Le gradient de la fonction potentiel peut alors être évalué ainsi :

$$\nabla f_w(\mathbf{p}) = J_{w^{-1}}^T(\mathbf{p}) \times \nabla f_A \circ w^{-1}(\mathbf{p})$$

où $J_{w^{-1}}^T(\mathbf{p})$ représente la transposée de la matrice jacobienne de la transformation inverse w^{-1} au point \mathbf{p} . Pour ces déformations, la forme close des expressions de w^{-1} et de $J_{w^{-1}}^T(\mathbf{p})$ peut être calculée. Les équations détaillées sont dans l'article de Barr [29].

La classification spatiale est accomplie en calculant d'abord la position $w^{-1}(\mathbf{p})$ du point de requête \mathbf{p} dans l'espace non déformé, puis en effectuant la requête de classification sur A comme suit :

$$c_w(\mathbf{p}) = c_A \circ w^{-1}(\mathbf{p})$$

La figure 3.11 montre une application de l'opération de torsion à deux cylindres implicites mélangés.

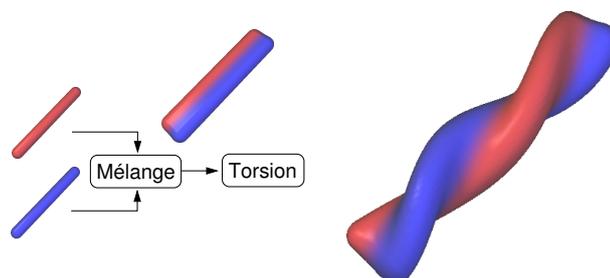


FIG. 3.11 – Opération de torsion appliquée à deux cylindres implicites mélangés.

3.2.5.2 Déformations libres

Les déformations libres ont été introduites pour la première fois par Sederberg et Parry [174] et ont été ensuite étendues par plusieurs auteurs [63, 47, 134]. L'application de déformations locales dans la formulation implicite n'est pas directe étant donné qu'il n'existe pas de moyen simple pour obtenir une formulation analytique de w^{-1} . Dans notre système, la fonction potentiel pour les opérations de type FFD est évaluée en utilisant une représentation maillée intermédiaire. L'algorithme procède comme suit :

1. Générer le maillage \mathcal{M}_A de A .
2. Déformer \mathcal{M}_A en appliquant la transformation w aux sommets de \mathcal{M}_A .
3. Évaluer la fonction potentiel en utilisant la fonction de distance au maillage déformé \mathcal{M}_A .

La figure 3.12 illustre notre outil de déformation libre, appliqué au modèle de bouteille de la figure 3.4. Cette approche implique que les nœuds FFD contiennent une représentation maillée de leur propre surface résultante. Les requêtes suivantes de potentiel, gradient ou classification spatiale sont effectuées directement sur ce maillage sans descendre dans le sous-arbre. Si A est une primitive simple, la fonction potentiel est évaluée en utilisant la fonction potentiel de A . Si la déformation locale s'étend à plusieurs primitives, alors leurs propriétés de mélange sont remplacées par une nouvelle fonction de distribution du potentiel dans l'espace associée au maillage. Par conséquent, ces nœuds devraient se situer au plus bas niveaux possibles dans l'arbre de manière à préserver ces propriétés propres à chaque primitive, si celles-ci se retrouvent impliquées dans d'autres opérations plus haut dans l'arbre.

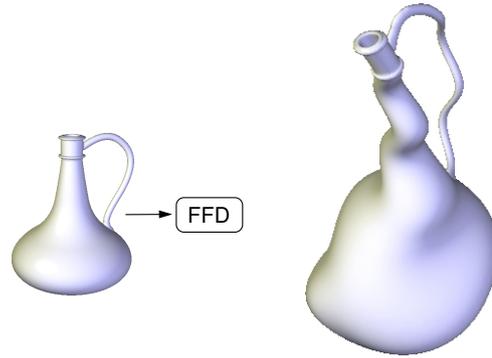


FIG. 3.12 – Déformation libre appliquée au modèle implicite de bouteille de la figure 3.4.

3.2.5.3 Transformations affines

Les transformations affines peuvent être appliquées aux surfaces implicites comme des déformations de l'espace. La fonction potentiel résultante peut alors être définie en utilisant la transformation inverse de la même manière que pour les déformations de Barr. Cependant, ceci nécessite d'évaluer w^{-1} pour toutes les requêtes sur le sous-arbre associé. En bénéficiant de la distributivité des transformations affines par rapport aux opérations booléennes et de mélange, Fox et al. [92] ont proposé dans le cadre du Blob-Tree [191] de retirer les nœuds de transformation affines pour les intégrer directement dans les paramètres des primitives. Dans leur méthode, le processus est bloqué par les nœuds de déformation. Dans notre système, nous avons étendu l'algorithme aux nœuds de mélange local et nous l'avons optimisé pour que les similarités euclidiennes, incluant les transformations rigides et les homothéties uniformes, puissent traverser les nœuds de déformation de Barr et de déformation libre. Pour ces trois opérations, les transformations affines sont transmises à la fois aux arguments et aux paramètres spatiaux de l'opération fille.

Soit $a\{N\}$ une transformation affine avec pour nœud fils N . Notre algorithme procède comme suit pour les différents types de nœud.

- Si N est une primitive, on a :

$$a\{N\} = N'$$

où N' est la primitive transformée.

- Si $o\{A, B\}$ est une opération booléenne ou de mélange global avec des nœuds fils A et B , on a l'équivalence suivante :

$$a\{o\{A, B\}\} = o\{a\{A\}, a\{B\}\}$$

- Si $o\{A, B, C\}$ est une opération de mélange local avec pour nœuds fils A , B et C , on a :

$$a\{o\{A, B, C\}\} = o\{a\{A\}, a\{B\}, a\{C\}\}$$

- Soit $FFD_{\mathcal{G}}\{N\}$ une opération de type FFD avec comme paramètre \mathcal{G} représentant la grille dans laquelle la surface associée au nœud fils N est plongée pour y être déformée. Si a est une similarité

euclidienne, nous avons :

$$a\{FFD_{\mathcal{G}}\{N\}\} = FFD_{\mathcal{G}'}\{a\{N\}\}$$

où \mathcal{G}' est la grille transformée.

- Soit $w_{\mathcal{F}}\{N\}$ une déformation de Barr avec pour nœud file N , appliquée dans un repère local \mathcal{F} . Si a est une similarité euclidienne, alors on a :

$$a\{w_{\mathcal{F}}\{N\}\} = w_{\mathcal{F}'}\{a\{N\}\}$$

où \mathcal{F}' représente le repère local transformé.

La figure 3.13 illustre la transmission d'une rotation à un objet complexe à travers un nœud FFD.

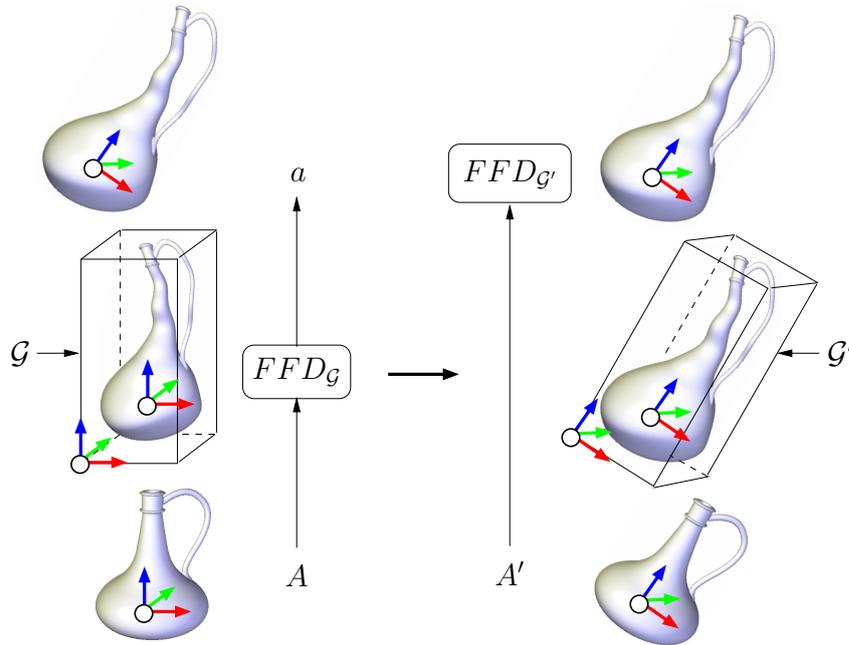


FIG. 3.13 – Transmission d'une rotation à travers un nœud FFD.

3.3 Maillage de l'HybridTree

La surface résultante d'un HybridTree peut être triangulée en utilisant les techniques classiques de maillage de surfaces implicites, grâce aux fonctions potentiel définies précédemment. Cependant, ces techniques nécessitent d'évaluer de nombreuses fois la fonction potentiel, ce qui est coûteux dans le cas général. En particulier, échantillonner la fonction potentiel pour une primitive de type maillage est une tâche qui coûte cher en temps, et qui n'est pas profitable si la surface maillée n'interagit que localement avec les autres primitives.

Pour un maillage efficace, nous avons développé une approche incrémentale qui préserve le plus

possible les surfaces maillées existantes et qui optimise la génération du maillage pour les différents types de nœuds en utilisant des algorithmes de maillage spécifiques. Les maillages locaux sont fusionnés au niveau des nœuds de mélange ou d'opérations booléennes pour former la surface finale. Les paragraphes suivants détaillent nos méthodes de maillage pour les primitives implicites à squelette et pour les différents types d'opérations de l'HybridTree.

3.3.1 Primitives implicites à squelette

Dans notre système, chaque primitive implicite est capable de générer automatiquement sa représentation maillée de manière très efficace pour un niveau de détail déterminé. Pour tous les types de squelettes, nous avons développé une procédure de maillage spécifique et optimisée qui produit en sortie un maillage manifold caractérisé par un échantillon uniforme et une connectivité régulière (fig. 3.14).

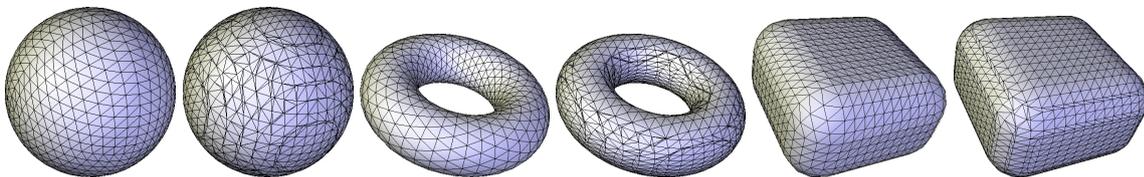


FIG. 3.14 – Maillages obtenus par maillage direct (à gauche) et résultats de l'algorithme des Marching Cubes (à droite) pour une même longueur d'arête. De gauche à droite : squelette point, cercle et boîte.

Comme mentionné précédemment, la surface de niveau T d'une primitive implicite à squelette peut être décrite par balayage d'une sphère de rayon constant $r_T = g^{-1}(T)$ le long du squelette. Cette isosurface correspond à une 2-variété et sera maillée si et seulement si $r_T > 0$. Pour la plupart des primitives, cette surface peut être décrite comme un patchwork de morceaux de surfaces simples tels que des portions de sphère, de cylindre, de plans, ou de disques, ce qui facilite le maillage direct. Après avoir identifié les différentes composantes pour une primitive donnée, nous échantillons chaque patch de manière itérative tout en établissant des relations de connectivité consistantes entre les points. Le niveau de détail est fixé par le choix d'une longueur d'arête maximale. Pendant le processus d'échantillonnage, nous profitons des symétries quand cela est possible. Par exemple, l'échantillonnage d'un octant de sphère suffit pour une primitive à squelette ponctuel ou sphérique [5]. Cette géométrie est alors répliquée par rapport aux axes ou aux plans de symétrie des primitives. Pour une primitive de type boîte, un octant de sphère, un quadrant de cylindre et un rectangle sont nécessaires. La table 3.2 présente des décompositions possibles pour différentes primitives à squelette.

En comparaison à un algorithme classique de maillage de surfaces implicites tels que les Marching Cubes [133], les temps de calcul montrent que cette approche accélère le processus jusqu'à 200 fois et produit jusqu'à 30% de triangles en moins, avec une forme bien mieux équilibrée. La figure 3.14 montre les maillages pour des primitives à squelette de type point, cercle et boîte produits par nos algorithmes, qui apparaissent de meilleure qualité que ceux obtenus par l'algorithme des [133] pour le même niveau de précision. Les temps de calcul et le nombre de triangles produits sont répertoriés dans la table 3.3 pour une résolution plus fine.

Squelette \ Surfaces	Sphère	Cylindre	Tore	Rectangle	Disque
Point	×				
Segment	×	×			
Cercle			×		
Boîte	×	×		×	
Cylindre	×	×			×

TAB. 3.2 – Décomposition en morceaux de surface simples de différentes primitives à squelette.

Primitive	Maillage direct		Marching Cubes		
	Temps	Nb. triangles	Temps	Nb. triangles	Nb. Cellules
Point	4	57 800	567	75 356	166 ³
Cercle	1	57 000	353	87 280	116 ³
Boîte	4	84 780	765	105 912	139 ³

TAB. 3.3 – Temps de maillage (en millisecondes) et nombre de triangles pour le maillage direct et l'algorithme des Marching Cubes pour une résolution fine (calculs effectués sur un Pentium IV 3.0 GHz avec 1 Go de RAM).

3.3.2 Opérations de mélange

Le maillage aux nœuds de mélange est généré en utilisant la représentation implicite de la surface localement. L'échantillonnage de l'isosurface est une tâche coûteuse qui nécessite de multiples évaluations de la fonction potentiel. Afin d'économiser du temps et préserver les maillages existants, le calcul des nouveaux points est restreint aux régions de mélange. Quand deux objets ne se recouvrent que peu, nous avons observé qu'il est plus intéressant de générer les maillages de ces objets avant de les combiner plutôt que de calculer directement un maillage global.

Les paragraphes suivants décrivent nos algorithmes de maillage pour les nœuds de mélange global et local. Dans le cas global, nous supposons dans un premier temps que les objets qui se mélangent sont associés avec des champs de potentiel positifs, ce qui constitue la situation la plus courante. Le cas des champs de potentiel négatifs sera traité dans un second temps.

3.3.2.1 Mélange global

Nous proposons une approche locale du maillage des nœuds de mélange global consistant à générer les maillages des deux nœuds fils, puis à construire le maillage dans les régions de mélange en utilisant l'algorithme des Marching Cubes [133]. Nous présentons deux types d'algorithmes qui diffèrent dans la manière dont ils traitent les différents nœuds de l'HybridTree. Le premier algorithme traverse récursivement la structure d'arbre, chaque nœud générant le maillage de son propre sous-arbre. Le second

algorithme est indépendant de la structure d'arbre et analyse les interactions entre les primitives.

Algorithme de maillage par nœud

Soient A et B deux objets qui se mélangent en un objet C . Nous supposons que A et B génèrent des champs de potentiel positifs. Si A et B sont seulement mélangés partiellement, alors le maillage de C est créé à partir des maillages \mathcal{M}_A et \mathcal{M}_B . Nous utilisons l'algorithme des Marching Cubes pour générer le maillage dans la région de mélange. Sinon, le maillage de C est complètement créé par l'algorithme des Marching Cubes. Dans les deux cas, les points de l'échantillon généré par Marching Cubes sont calculés en évaluant la fonction potentiel locale f_{A+B} .

Pour déterminer l'approche la plus favorable, nous estimons à quel point les modèles A et B se recouvrent, i.e. quelle proportion de leur volume est impliquée dans la région de mélange. Rappelons qu'entre A et B , le mélange se produit dans les régions de l'espace où les fonctions f_A et f_B sont positives. Ces régions de mélange sont incluses dans l'intersection de \mathcal{B}_A et \mathcal{B}_B . Nous introduisons un rapport ρ appelé *taux de recouvrement* tel que $0 \leq \rho \leq 1$, qui est calculé comme suit :

$$\rho = \frac{V_{A \cap B}}{\min(V_A, V_B)}$$

où V_A , V_B et $V_{A \cap B}$ désignent les volumes des boîtes \mathcal{B}_A , \mathcal{B}_B et $\mathcal{B}_A \cap \mathcal{B}_B$ respectivement. Soit $\rho_0 \in [0, 1]$ un seuil fixé. Sans perte de généralité, supposons que $V_A \leq V_B$. Notre algorithme procède alors de la manière suivante :

1. Si $\rho > \rho_0$, alors :
 - (a) Si $\frac{V_A}{V_B} \leq \rho_0$, alors appliquer l'algorithme des Marching Cubes dans la plus petite boîte englobante de $\mathcal{B}_A \cup \mathcal{B}_B$.
 - (b) Sinon :
 - Créer le maillage \mathcal{M}_B de B .
 - Supprimer les triangles de \mathcal{M}_B qui ont au moins un sommet \mathbf{p}_i tel que $\mathbf{p}_i \in \mathcal{B}_A$.
 - Appliquer l'algorithme des Marching Cubes dans la boîte \mathcal{B}_A .
2. Sinon :
 - (a) Créer les maillages de \mathcal{M}_A et \mathcal{M}_B de A et B respectivement.
 - (b) Supprimer les triangles de \mathcal{M}_A et \mathcal{M}_B qui ont au moins un sommet \mathbf{p}_i tel que $\mathbf{p}_i \in \mathcal{B}_A \cap \mathcal{B}_B$.
 - (c) Appliquer l'algorithme des Marching Cubes dans la boîte $\mathcal{B}_A \cap \mathcal{B}_B$.

Nous avons recours à un algorithme de couture après chaque phase de Marching Cubes local de manière à raccorder les bords des différents maillages générés pour produire un maillage fermé. Notre algorithme crée de nouveaux triangles à partir des arêtes des bords. Pour chaque paires de contours, nous identifions les plus proches voisins de part et d'autre, puis ces sommets sont connectés entre eux. Cette technique donne de bons résultats si les maillages \mathcal{M}_A et \mathcal{M}_B ont des triangles de taille presque identique et si les bords qui correspondent sont très proches les uns des autres. Pour que la seconde

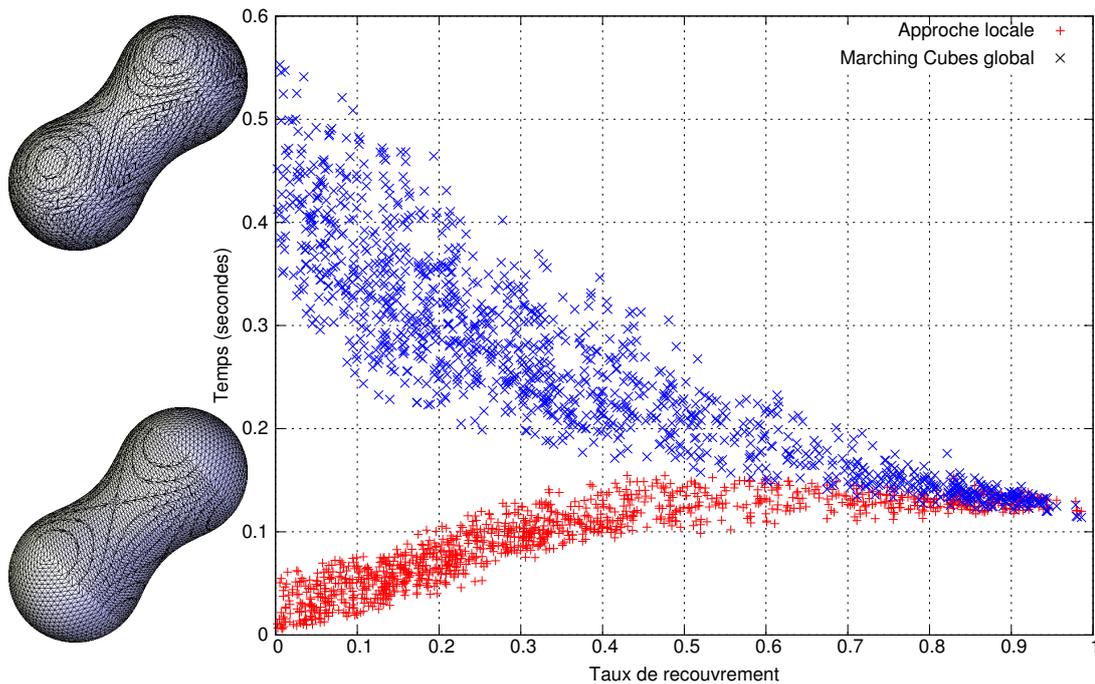


FIG. 3.15 – Temps de maillage en fonction du taux de recouvrement pour 10^3 requêtes de maillage sur deux sphères implicites mélangées avec des positions variables dans l'espace. À gauche, l'image du haut montre le résultat d'un Marching Cubes global et l'image du bas montre le résultat de l'approche locale.

condition soit vérifiée, les triangles de \mathcal{M}_A et/ou \mathcal{M}_B qui intersectent la boîte englobante de la région de mélange sont découpés autour de cette boîte. Quand la longueur des arêtes qui se font face est trop différente, les triangles des bords sont raffinés de manière à rendre compatible l'échelle des arêtes des bords.

Avant de lancer le processus de maillage, nous calculons le rapport ρ pour les nœuds de mélange terminaux en faisant remonter l'information en haut de l'arbre. Par souci d'efficacité, les nœuds de mélange consécutifs le long d'une même branche dont les fils se recouvrent de manière importante sont regroupés de manière à les traiter en une seule passe de Marching Cubes au plus haut niveau possible de la branche. Ce processus de regroupement le long d'une branche s'arrête dès qu'un nœud autre qu'un nœud de mélange global est rencontré.

L'utilisateur peut fournir une valeur pour ρ_0 ou directement spécifier la méthode à utiliser pour chaque opération de mélange dans un même arbre. Nous choisissons $\rho_0 = 0.5$ comme seuil par défaut, qui donne des résultats satisfaisants dans la plupart des cas. La motivation de ce choix est illustrée par la figure 3.15. Nous avons comparé le temps nécessaire pour mailler deux sphères implicites en utilisant notre approche locale et un Marching Cubes global pour des positions aléatoires des deux primitives. On peut observer sur le diagramme que notre approche locale est jusqu'à cinq fois plus rapide quand le volume de recouvrement est inférieur à 0.5. Le taux de recouvrement mesure cependant le recouvrement entre les boîtes englobantes des objets, et n'est pas nécessairement le reflet exact du recouvrement de ces objets, en particulier si le volume des objets est faible relativement à celui des boîtes. Il s'agit cependant

d'un compromis intéressant en pratique.

En s'appuyant sur l'analyse d'une hiérarchie de boîtes englobantes, cette approche est particulièrement simple et systématique. Cependant, ses performances dépendent clairement de la structuration de l'HybridTree. À cause d'une structure d'arbre binaire non optimale, certaines régions de la surface résultante peuvent en effet être amenées à être maillées plusieurs fois par Marching Cubes inutilement, comme par exemple dans le cas de primitives qui se mélangent en étant localisées dans des sous-arbres différents. Cette limite peut être compensée par une restructuration de l'arbre, mais ce type d'optimisation est coûteuse à mettre en œuvre [27] et ne permet pas d'améliorer toutes les configurations, comme illustré par le modèle de la figure 3.16. Dans cet exemple, le nœud de mélange serait globalement maillé par Marching Cubes alors que le mélange ne se produit que localement.

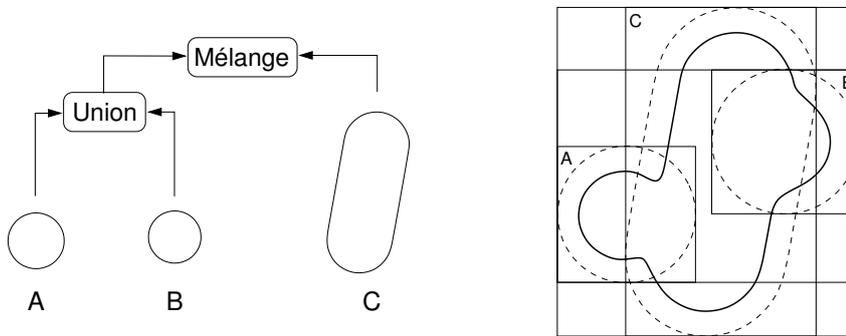


FIG. 3.16 – Une configuration d'HybridTree pour laquelle notre algorithme de maillage nœud pour les nœuds de mélange n'est pas optimal.

La difficulté est en fait intrinsèque à la représentation sous la forme d'un arbre binaire, qui ne permet pas de décrire explicitement les interactions entre plus de deux primitives. Pour faire face à des structures d'arbre binaire non optimales, nous proposons d'étendre notre premier algorithme dans l'esprit de l'approche par décomposition de l'espace introduite par Fox et al. dans [92] pour le modèle du BlobTree. Cette approche se focalise sur la manière dont les primitives interagissent dans l'espace plutôt que sur la structure globale de l'arbre.

Algorithme de maillage par primitive

Nous distinguons deux types de régions de l'espace : les régions \mathcal{R}_P qui sont influencées par une seule primitive (fig. 3.17(a), gris clair) et les régions \mathcal{R}_+ de mélange (fig. 3.17(a), gris foncé). Notre objectif est de mailler les primitives en utilisant le maillage direct dans les régions \mathcal{R}_P et d'appliquer l'algorithme des Marching Cubes en une seule fois dans les régions \mathcal{R}_+ . Dans ce but, l'HybridTree est plongé dans une grille régulière et nous mettons en œuvre un algorithme de Marching Cubes modifié pour calculer l'échantillon de points d'échantillon dans les boîtes englobantes des régions de mélange.

Soit $\{P_j\}$, $j = 1..n$ l'ensemble des primitives d'un arbre de construction donné, avec pour boîtes

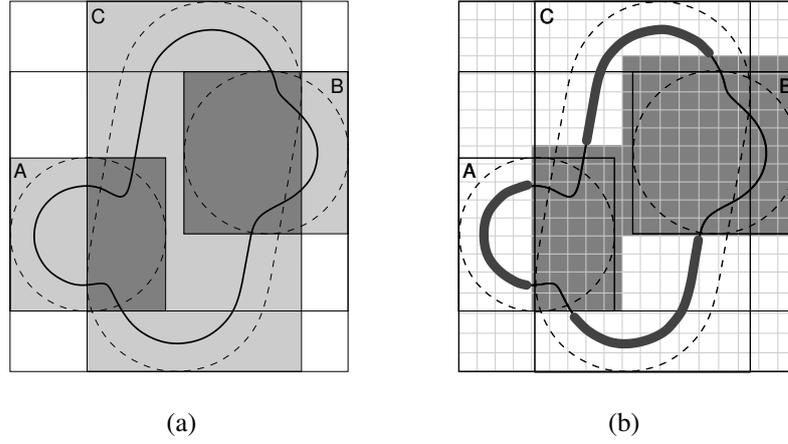


FIG. 3.17 – Notre méthode de maillage local primitive optimisée.

englobantes \mathcal{B}_j . Nous définissons le taux de recouvrement ρ_i pour une primitive P_i comme suit :

$$\rho_i = \frac{V_O(\mathcal{B}_i)}{V(\mathcal{B}_i)}$$

où $V(\mathcal{B}_i)$ est le volume de la boîte \mathcal{B}_i et $V_O(\mathcal{B}_i)$ le volume de \mathcal{B}_i qui est partagé avec les boîtes englobantes des autres primitives, i.e. :

$$V_O(\mathcal{B}_i) = V \left(\bigcup_{j \neq i} \mathcal{B}_i \cap \mathcal{B}_j \right)$$

où $V(\bigcup_{k=1}^n \mathcal{B}_k)$ est calculé en utilisant la formule d'inclusion-exclusion :

$$V \left(\bigcup_{k=1}^n \mathcal{B}_k \right) = \sum_{l=1}^n (-1)^{l+1} \sum_{m_1 < m_2 < \dots < m_l} V(\mathcal{B}_{m_1} \cap \mathcal{B}_{m_2} \cap \dots \cap \mathcal{B}_{m_l})$$

Soit \mathcal{G} une grille régulière de l'espace dans laquelle est plongé l'HybridTree. Toutes les cellules de la grille sont d'abord initialisées à "0". L'algorithme procède comme suit pour chaque primitive P_i :

1. Calculer l'ensemble $\{\mathcal{B}_i \cap \mathcal{B}_j\}, j \neq i, \mathcal{B}_i \cap \mathcal{B}_j \neq \emptyset$, tel que le premier ancêtre commun A de P_i et P_j est un nœud de mélange et P_j n'est pas un descendant droit d'une opération de différence localisée en dessous de A .
2. Dilater \mathcal{B}_i et toutes les boîtes $\mathcal{B}_i \cap \mathcal{B}_j, j \neq i$ de manière à les aligner sur la grille \mathcal{G} .
3. Calculer ρ_i .
4. Si $\rho_i < \rho_0$, alors :
 - (a) Créer le maillage \mathcal{M}_i de P_i .
 - (b) Supprimer les triangles de \mathcal{M}_i ayant au moins un sommet \mathbf{p}_i tel que $\mathbf{p}_i \in \bigcup_{j \neq i} \mathcal{B}_j$.
 - (c) Marquer "1" toutes les cellules de \mathcal{G} qui se trouvent dans $\bigcup_{j \neq i} \mathcal{B}_j$.

5. Sinon, marquer "1" toutes les cellules de \mathcal{G} qui se trouvent dans \mathcal{B}_i .

Notre algorithme de Marching Cubes modifié calcule alors des points d'échantillon le long des arêtes des cellules marquées "1" et triangule ces cellules.

En utilisant cette méthode, plus de primitives implicites sont maillées par la méthode directe et davantage de parties de maillage existantes sont préservées. Dans la figure 3.17(b), les régions maillées en utilisant la méthode directe sont représentées avec un contour gras. La figure 3.18 montre les maillages obtenus selon trois approches différentes. Dans ce modèle, plusieurs primitives implicites sont localement mélangées avec le modèle IGEA, amenant à une configuration similaire à celle de la figure 3.16. Les maillages ont été obtenus en appliquant l'algorithme des Marching Cubes global (à gauche), puis notre algorithme de maillage par nœud (au centre), et finalement notre algorithme de maillage par primitive (à droite). Les temps de calcul sont répertoriés dans la table 3.4, qui montre aussi le nombre de points d'échantillon calculés sur la surface à travers la technique des Marching Cubes. Pour ce modèle, notre algorithme de maillage par primitive est deux fois plus rapide que notre algorithme de maillage par nœuds.

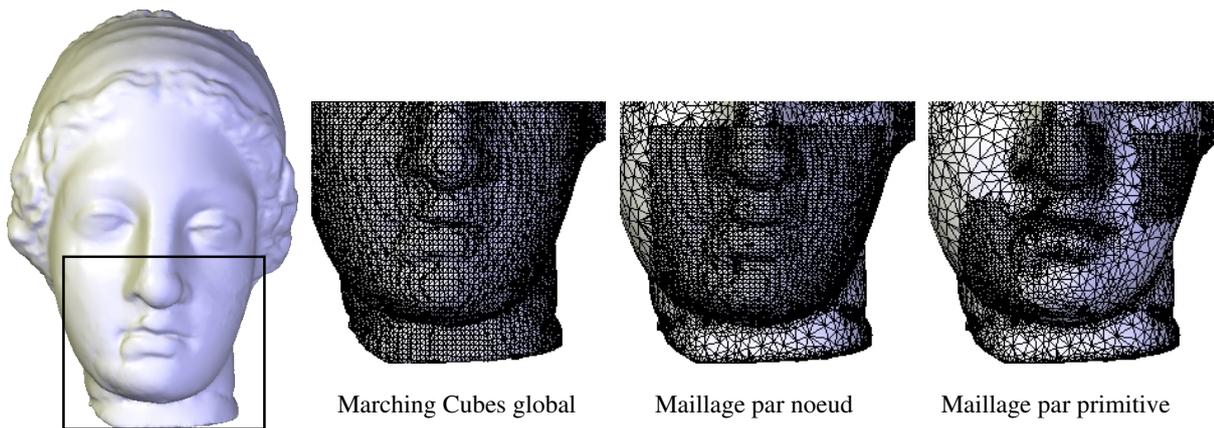


FIG. 3.18 – Plusieurs résultats de maillages pour le modèle partiellement restauré d'IGEA de la figure 3.24.

Méthode	Temps maillage	Nb. points M.C.	Nb. triangles
M.C. global	1 533.02	54 420	108 836
Algo. par nœud	19.82	8 299	61 980
Algo. par primitive	9.67	2 592	58 458

TAB. 3.4 – Temps de maillage (en secondes), nombre de nouveaux points d'échantillon calculés par la méthode des Marching Cubes et nombre de triangles pour le Marching Cubes global, l'algorithme de maillage par nœuds et l'algorithme de maillage par primitive (calculs effectués sur un Pentium IV 3.0 GHz avec 1 Go de RAM).

Le calcul des boîtes d'intersection entre toutes les paires de primitives est accompli en temps $O(n^2)$, où n est le nombre de primitives dans l'arbre. Pour une primitive donnée, le taux de recouvrement

est évalué en temps $O(n^2)$ dans le pire cas. Par conséquent, les performances de notre algorithme de maillage par primitive peuvent décliner sur un ensemble de primitives qui se mélangent en se recouvrant beaucoup. Cependant, en pratique, le nombre de primitives qui contribuent effectivement au champ de potentiel final en un point donné de l'espace est généralement petit comparé au nombre total de primitives dans un modèle particulier.

Notre méthode de maillage par primitive nécessite aussi de stocker une grille de taille m^3 avec seulement un bit par cellule. Pour une grille avec 300^3 cellules, ce qui était le maximum dans nos tests, ceci représente moins de 3.5 Mo de mémoire. Le temps de parcours l'ensemble des cellules est négligeable par rapport au processus de maillage. Si davantage de précision est nécessaire, ou bien si la mémoire est une ressource critique, alors l'approche nœuds peut être plus profitable.

3.3.2.2 Maillage des nœuds de mélange négatif

Supposons maintenant que A génère un champ de potentiel positif et B un champ de potentiel négatif. Dans ce cas, notre algorithme procède comme suit :

1. Créer le maillage \mathcal{M}_A de A .
2. Supprimer les triangles de \mathcal{M}_A avec au moins un sommet p_i tel que $p_i \in \mathcal{B}_B$.
3. Appliquer l'algorithme des Marching Cubes dans \mathcal{B}_B et invoquer l'algorithme de couture pour fermer le maillage.

3.3.2.3 Maillage des nœuds de mélange local

Ici, nous supposons que A et B génèrent des champs de potentiel positifs. Le maillage des nœuds de mélange local est réalisé comme suit :

1. Créer les maillages \mathcal{M}_A et \mathcal{M}_B de A et B respectivement.
2. Construire le maillage \mathcal{M}_D de l'union D entre A et B .
3. Supprimer les triangles de \mathcal{M}_D qui ont au moins un sommet p_i tel que $p_i \in \mathcal{B}_A \cap \mathcal{B}_B$ ou $p_i \in \mathcal{B}_A \cap \mathcal{B}_C$.
4. Appliquer l'algorithme des Marching Cubes dans \mathcal{B}_R et invoquer l'algorithme de couture pour fermer le maillage.

3.3.3 Opérations booléennes

La construction du maillage résultant d'une opération booléenne est réalisée comme dans les modélisateurs B-Rep standards. Notre approche tire parti de la représentation duale implicite/maillage de l'HybridTree. Nous utilisons la représentation implicite des nœuds fils pour réaliser efficacement les tests de classification spatiale. L'algorithme peut être écrit comme suit pour les opérations d'union, d'intersection ou de différence :

1. Créer les maillages \mathcal{M}_A et \mathcal{M}_B de A et B respectivement.
2. Si \mathcal{B}_A et \mathcal{B}_B se recouvrent, alors calculer le maillage résultant en utilisant les fonctions de classification spatiale de A et de B .

Pour déterminer si deux triangles se recouvrent et les découper correctement, nous utilisons le test de recouvrement triangle-triangle robuste proposé par Guigue et Devillers [106].

3.3.4 Opérations de déformation

Nous créons d'abord le maillage \mathcal{M}_A du nœud fils A . La déformation est alors appliquée au maillage \mathcal{M}_A en appliquant la transformation aux coordonnées des sommets pour obtenir le maillage déformé.

Les translations, rotations et homothéties uniformes préservent la forme des triangles du maillage, tandis que les homothéties non uniformes ou les torsions, écrasement ou courbure peuvent conduire à des triangles aplatis. Dans ce cas, nous appliquons un simple processus de remaillage local par contraction d'arêtes [111] ou insertion de sommets pour améliorer la forme des triangles.

3.4 Détails d'implantation

Nous avons implanté un prototype de notre modèle en C++ sur un système Linux. La figure 3.19 montre le diagramme d'héritage des nœuds de l'HybridTree, où le nom des classes abstraites apparaît en italique. Le diagramme ne présente pas toutes les primitives pour des raisons de clarté. Les fonctions implantant les algorithmes d'évaluation des différentes requêtes sont virtuelles pour les classes abstraites, et spécialisées pour les différents types de nœuds.

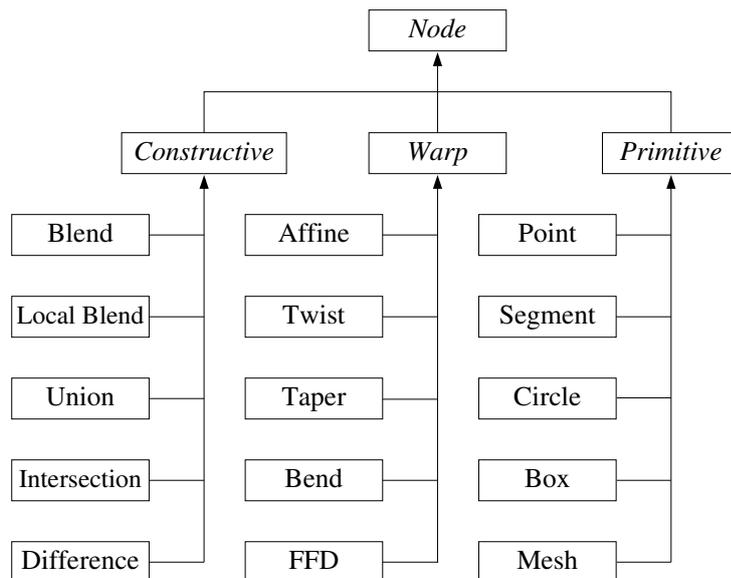


FIG. 3.19 – Diagramme d'héritage des classes de l'HybridTree.

Nous avons implanté les maillages en utilisant une structure d'arêtes orientées, comme proposé par Campagna et al. [48]. Bien que cette structure soit relativement redondante, elle permet une localisation rapide des arêtes des bords des maillages, ce qui est utile pour les fréquentes étapes de couture.

3.5 Résultats et discussion

Dans cette section, nous présentons des modèles complexes créés à l'aide de l'HybridTree. La Table 3.5 récapitule les temps de maillage de différents modèles hybrides, ainsi que le nombre de triangles générés. Les temps de prétraitement fournis prennent en compte le temps nécessaire à la construction de la hiérarchie de boîtes englobantes des modèles maillés et l'initialisation de la grille pour l'algorithme des Marching Cubes lorsque l'algorithme de maillage par primitive a été employé pour les nœuds de mélange local. Ces temps n'incluent pas le temps nécessaire à la reconstruction de modèles maillés depuis un échantillon de points lorsque de tels modèles ont été utilisés. Pour le modèle IGEA que nous avons utilisé, le temps de reconstruction sélective pour produire le modèle maillé a été de 15 secondes. Les mesures de temps ont été effectuées sur une machine équipée d'un processeur Pentium IV 3.0 GHz avec 1 Go de RAM.

3.5.1 Modélisation libre

La méduse ailée Les figures 3.1 et 3.20 montrent le résultat d'opérations booléennes et de mélanges appliquées à des modèles implicites, des maillages et un échantillon de points. Le modèle original de la MÉDUSE (fig. 3.1(a)) est un modèle implicite construit à partir de 250 primitives de type spline mélangées ensemble, qui fait partie de notre bibliothèque de modèles. Le corps a été tout d'abord mélangé avec un maillage du modèle IGEA (62 323 triangles) qui a été automatiquement reconstruit à partir de l'échantillon de points présenté dans la figure 3.3, ainsi qu'avec les ailes de la VICTOIRE DE SAMOTHRACE (16 340 triangles). Le processus de maillage du modèle hybride commence par créer un maillage du modèle implicite de la méduse ailée. Étant donné que les primitives implicites se recouvrent beaucoup, l'algorithme des Marching Cubes est appliqué. Le maillage résultant consiste de 121 524 triangles, et est généré en 6 secondes. La tête a ensuite été retirée en utilisant une opération booléenne de différence avec une primitive implicite à squelette ponctuel, et le corps a été mélangé avec le modèle IGEA en utilisant notre méthode de maillage local par nœud. Les ailes ont été extraites du modèle maillé de la VICTOIRE DE SAMOTHRACE en calculant l'intersection de ce modèle avec une boîte implicite. Les ailes et le modèle modifié de la MÉDUSE ont été finalement mélangés en utilisant la méthode locale de maillage par nœud.

Le vase Le vase de la figure 3.21 a été créé en utilisant des opérations de mélange. L'intérieur du modèle IGEA a été creusé en utilisant un champ de potentiel négatif généré par une primitive implicite de type cylindre. Les poignées ont été créées à partir de deux primitives implicites de type cercle, qui ont été ajoutées en utilisant l'opération de mélange local par primitive.

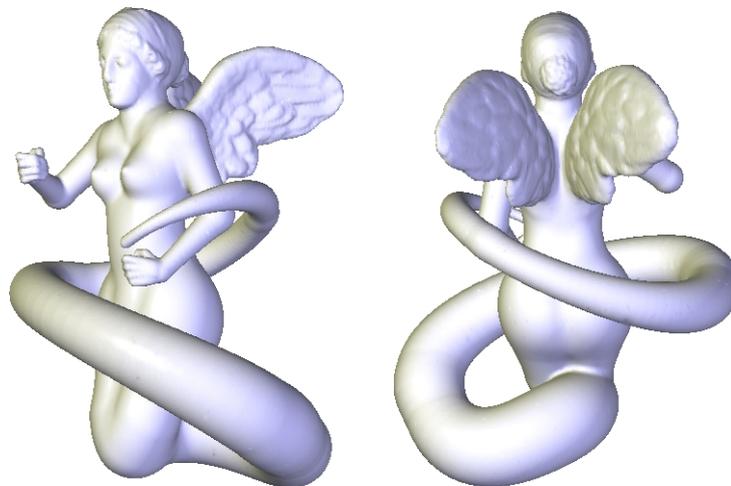


FIG. 3.20 – La méduse ailée de la figure 3.1.



FIG. 3.21 – Un vase créé à partir du modèle IGEA.

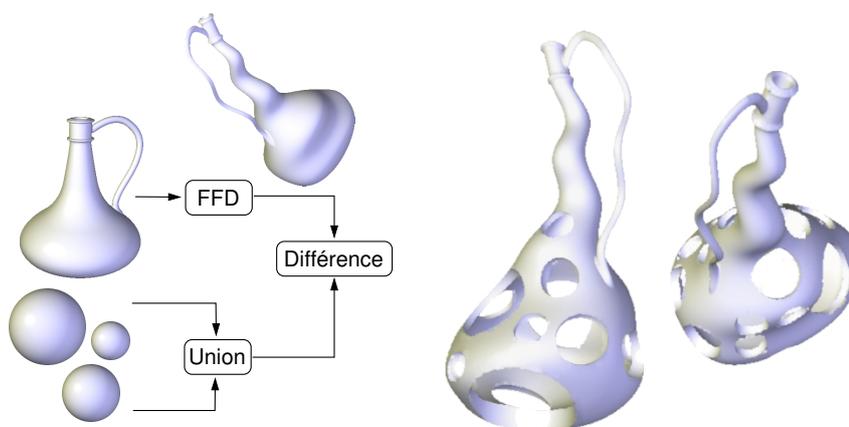


FIG. 3.22 – Une bouteille trouée.

Figure	Temps prétraitement	Temps maillage	Nb. triangles
3.1, 3.20	4.63	63.85	171 562
3.9	1.72	48.16	105 467
3.21	0	14.02	121 271
3.22	1.97	21.24	94 862
3.23	6.85	56.35	269 698
3.24	1.58	9.12	58 458

TAB. 3.5 – Temps de prétraitement et de maillage (en secondes) et nombre de triangles obtenus pour plusieurs modèles hybrides complexes.

La bouteille trouée La bouteille de la figure 3.22 a été créée à partir du modèle implicite de bouteille de la figure 3.4 qui comprend 5 primitives à squelettes complexes. Nous avons tout d’abord appliqué notre outil de déformation FFD, qui a nécessité de mailler la modèle de la bouteille. Ensuite, 12 trous ont été créés à l’aide d’opérations de différence booléenne avec des primitives implicites ponctuelles. Le maillage final a été obtenu en 21 secondes.

La Victoire La figure 3.23 montre un modèle de statue construit à partir du modèle maillé de la VICTOIRE DE SAMOTHRACE (187 072 triangles), qui présente la particularité de ne plus avoir de tête et de bras. Nous avons reconstitué un modèle complet en y ajoutant tout d’abord les bras du modèle implicite original de la MÉDUSE. Nous les avons mélangés avec le modèle de la VICTOIRE DE SAMOTHRACE en utilisant des sphères implicites au niveau de chaque épaule. Nous avons ensuite mélangé la surface résultante localement avec le modèle IGEA pour la tête en utilisant un cylindre implicite placé autour du cou. Nous avons finalement complété notre modèle de Victoire en lui ajoutant dans la main droite un bâton de berger représenté par des primitives implicites. Le maillage final a été obtenu en 56 secondes.

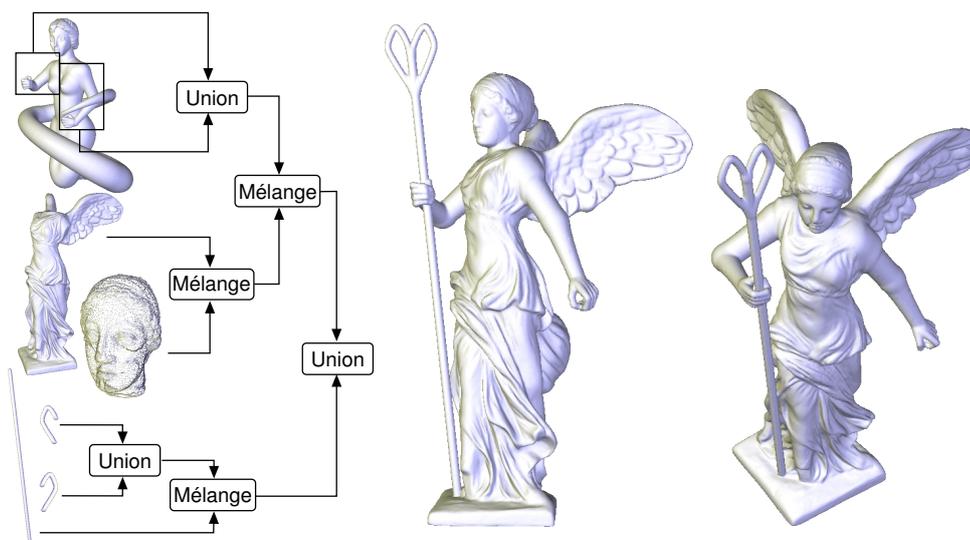


FIG. 3.23 – Un modèle de victoire.

3.5.2 Restauration virtuelle

Notre modèle hybride est adapté à la modélisation de formes complexes à partir de modèles existants ou à partir de primitives simples. Il peut être également avantageusement exploité dans un contexte de sauvegarde du patrimoine historique et culturel. L'HybridTree peut être utilisé efficacement pour simuler la restauration ou des phénomènes naturels [136] sur des œuvres d'art numérisées, et la structure hiérarchique maintient naturellement un historique des opérations, ce qui est utile à des fins d'archivage.

Restauration d'Igea La figure 3.24 illustre un processus de restauration virtuelle du modèle IGEA en utilisant l'HybridTree. Nous avons souhaité combler les fissures sur la droite du menton et sur la joue gauche, ainsi que reconstituer le nez, comme un spécialiste aurait pu le faire. Nous avons utilisé nos outils de mélange pour simuler l'utilisant d'un ciment de manière intuitive et réaliste.

Nous avons disposé manuellement les primitives implicites de type spline le long que chaque fissure et une primitive ponctuelle au bout du nez. Les valeurs des paramètres de la fonction potentiel ont aussi été réglées à la main pour chaque primitive. Les primitives ont ensuite été mélangées avec le modèle IGEA pour obtenir le modèle maillé final. Nous avons construit un sous-arbre indépendant pour l'ensemble des primitives au niveau du menton, et un autre pour le nez. Les premières ont été maillées en utilisant l'algorithme des Marching Cubes, les primitives se recouvrant beaucoup. Le maillage résultant a été mélangé avec le modèle IGEA en utilisant la méthode locale par primitive. La même approche a été utilisée pour le nez. Le maillage final a été obtenu en environ 10 secondes.

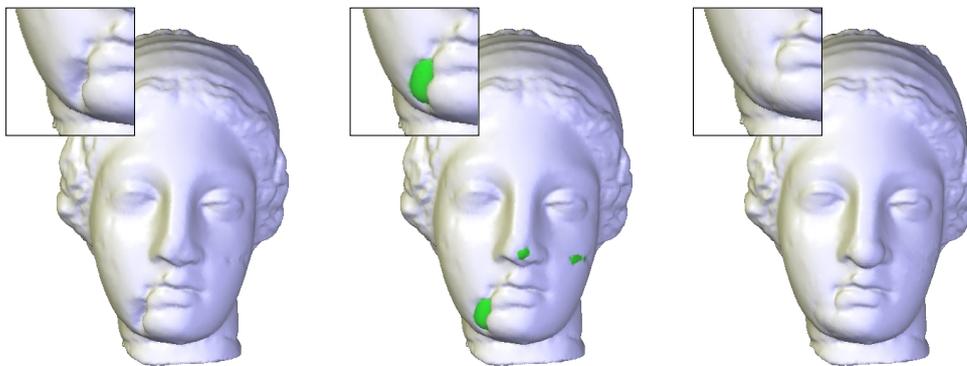


FIG. 3.24 – Les étapes de restauration du modèle IGEA. Le maillage initial apparaît à gauche. Au centre, de la matière a été ajoutée pour combler des fissures. À droite est montré un maillage du modèle partiellement restauré.

3.5.3 Discussion

Performances Notre système peut manipuler des primitives implicites à squelette complexes et des maillages polygonaux jusqu'à 25 000 triangles en temps interactif. Les déformations FFD ainsi que les opérations locales de mélange peuvent être accomplies interactivement pour des résolutions qui ne sont

pas trop fines. Les opérations booléennes combinant de petites primitives implicites ou de petits maillages en comparaison de l'échelle de l'objet final peuvent aussi être réalisées en temps interactif.

Les étapes de conversion entre maillages et surfaces implicites est un facteur limitant les performances du système. L'évaluation de la fonction potentiel générée par un maillage en un point donné de l'espace reste coûteux en temps, malgré notre technique d'accélération. Des expériences démontrent qu'une requête d'évaluation de la fonction potentiel sur un maillage complexe peut avoir un coût en temps qui est plusieurs centaines de fois celui de la même requête accomplie sur une primitive ponctuelle.

Pour la modélisation interactive ou l'animation, l'évaluation pourrait être accélérée en échantillonnant le champ de potentiel sur une grille régulière et en stockant les valeurs de potentiel, comme proposé par Schmidt et al. pour le modèle du BlobTree [171]. Une approximation de la surface peut être calculée par interpolation tricubique. Cette approche implique cependant un surcoût en mémoire et une possible perte d'information géométrique et topologique du fait de la résolution limitée de la grille. Une approche adaptative, à la manière des ADFs [93] pourrait fournir de bons résultats, mais également au prix d'une gestion plus complexe du champ de potentiel échantillonné.

Stockage La structure d'HybridTree permet de réduire de manière significative le coût de stockage de modèles complexes en mémoire. Contrairement aux modèles de type level set [148] ou ADFs [93], nous n'avons pas besoin d'échantillonner les objets l'intégralité des objets sur une grille. L'utilisation de primitives implicites à squelettes complexes permet de créer des formes complexes avec une représentation très compacte. Par exemple, le modèle de MÉDUSE qui apparaît dans la figure 3.1 a été créé en mélangeant 250 primitives à squelette de type spline. La représentation implicite correspondante consomme moins de 64 kilo-octets en mémoire.

Contrôle de la forme La capacité de combiner des modèles maillés et des surfaces implicites à squelette dans un système cohérent n'étend pas seulement la gamme des modèles qui peuvent être créés, mais elle nous permet aussi d'avoir un contrôle fin de la forme des objets.

La représentation implicite permet de réaliser des opérations de mélange sur des maillages de géométrie et de topologie quelconque, ce qui n'est pas le cas d'autres méthodes spécifiques de fusion de maillage comme celle de Singh et Parent [178] ou celle de Kanai et al. [118]. Celles-ci sont en effet plus restrictives sur la topologie des surfaces qui peuvent être fusionnées. Dans notre système, l'utilisateur peut contrôler de manière simple la manière dont se mélange les objets entre eux, de manière indépendante pour chaque objet. La représentation implicite fournit aussi le moyen de créer des mélanges négatifs entre les objets, ce qui est important pour simuler des effets de creusement. Enfin, notre outil FFD permet d'appliquer des déformations de manière intuitive et non restrictive à nos modèles hybrides.

Conclusions et perspectives

Dans cette seconde partie, nous avons étudié un modèle géométrique hybride dans lequel plusieurs représentations de surface coopèrent dans le but d'effectuer des opérations d'édition de manière non restrictive en utilisant la représentation la plus adaptée pour chacune. Ce modèle, que nous appelons l'HybridTree, est un arbre de construction pouvant prendre en entrée des surfaces implicites à squelette, des maillages et des modèles représentés par des échantillons de points. Ces modèles sont assemblés au moyen d'opérations booléennes, de mélange ou de déformation, incluant des déformations locales. Le système d'évaluation actuel de l'HybridTree est conçu pour exploiter les avantages complémentaires des surfaces implicites et des maillages en mettant en œuvre des mécanismes de conversion locale. Les opérations de modélisation sont réalisées dans la représentation la plus adaptée, de manière transparente pour l'utilisateur. La représentation maillée est utile pour la visualisation rapide et les déformations libres, et la représentation implicite se prête aux opérations booléennes et de mélange, de par sa nature volumique. L'HybridTree est évalué au travers de requêtes d'évaluation de la fonction potentiel, de gradient, de classification spatiale et de maillage qui sont optimisées pour chaque type de nœud.

Nous avons illustré les capacités de notre modèle à travers plusieurs exemples de modélisation d'objets complexes mettant en évidence ses aptitudes en termes de diversité d'opérations et de contrôle local. Sa représentation est peu coûteuse à stocker et permet de bénéficier de la compacité du modèle des surfaces implicites pour les primitives à squelette. Cet arbre maintient naturellement un historique de construction conservant les représentations initiales des modèles en entrée, ce qui est un avantage pour remonter en cause des opérations.

Pour autant, de nombreuses améliorations sont souhaitables afin de favoriser le développement de notre modèle. Tout d'abord, des conversions successives liées à l'application de FFD peuvent entraîner une accumulation d'erreurs d'approximation de la surface. Déformer localement un sous-arbre en évitant sa conversion complète en maillage est un problème ouvert dans notre modèle, qui ne l'est pas avec un modèle de surface de type MLS. L'intégration d'un tel modèle dans l'HybridTree pourrait être favorable lors de l'application de FFD sur des surfaces lisses, à condition de savoir le combiner de manière cohérente avec les autres représentations.

Nous souhaitons pouvoir gérer automatiquement des niveaux de détail dans l'HybridTree. Sa structure hiérarchique s'y prête particulièrement bien, pouvant de plus hériter des capacités de représentation multiéchelle des maillages et des surfaces implicites. Nous pensons qu'il serait possible de combiner les primitives implicites à squelette avec des niveaux de détail, comme proposé par Angelidis et Cani [19] avec des squelettes définis à partir de courbes ou de surfaces de subdivision. La prise en compte de ni-

veaux de détail dans l'HybridTree pourrait en outre permettre d'améliorer la localité des conversions, en cernant plus précisément les régions affectées par des opérations que par le simple recours à des boîtes englobantes.

L'HybridTree ne dispose actuellement que d'une interface déclarative. Il serait intéressant de développer une interface de modélisation interactive, qui nous permettrait d'explorer d'autres pistes de recherche. La question d'assurer efficacement la réversibilité des opérations en cas d'annulation est particulièrement intéressante dans ce cadre hybride.

Pour finir, l'HybridTree peut être considéré comme un modèle ouvert pouvant intégrer de nouvelles représentations par définition de passerelles avec les modèles existants. La représentation la plus adaptée pour chaque opération pourrait être alors choisie selon des critères plus précis, par exemple en déterminant le modèle qui permet de produire le résultat de meilleure qualité, ou celui qui fournit le résultat le plus rapidement. D'autres critères pourraient être la progressivité, ou le point de vue dans une scène, de manière à n'effectuer que les opérations utiles.

Bibliographie

- [1] <http://www.cgal.org>. 64
- [2] A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2) :239–256, 1992. 3
- [3] A. Adamson and M. Alexa. Approximating and intersecting surfaces from points. In *Proc. Symposium on Geometry Processing*, pages 245–254, 2003. 21, 31
- [4] U. Adamy, J. Giesen, and M. John. Surface reconstruction using umbrella filters. *Computational Geometry*, 21(1–2) :63–86, 2002. 15
- [5] J. Ahn. Fast Generation of Ellipsoids. *Graphics Gems V*, pages 179–190, 1995. 140
- [6] M. Ahn, S. Lee, and H.-P. Seidel. Connectivity transformation for mesh metamorphosis. In *Proc. Eurographics/ACM SIGGRAPH Symposium on Geometry Processing*, pages 77–84, 2004. 114
- [7] S. Akkouche and E. Galin. Adaptive Implicit Surface Polygonization using Marching Triangles. *Computer Graphics Forum*, 20(2) :67–80, 2001. 119
- [8] M. Alexa. Recent Advances in Mesh Morphing. *Computer Graphics Forum*, 21(2) :173–196, 2002. 114
- [9] M. Alexa, J. Behr, D. Cohen-Or, S. Fleishman, D. Levin, and C. T. Silva. Point set surfaces. In *Proc. IEEE Visualization Conference*, pages 21–28, 2001. 29, 30
- [10] P. Alliez and C. Gotsman. Recent Advances in Compression of 3D Meshes. *Advances in Multiresolution for Geometric Modelling*, pages 3–26, 2005. 4
- [11] P. Alliez, G. Ucelli, C. Gotsman, and M. Attene. Recent advances in remeshing of surfaces. *State-of-the-art report of the AIM@SHAPE EU network*, 2005. 117
- [12] N. Amenta. The crust algorithm for 3d surface reconstruction. In *Proc. Symposium on Computational Geometry*, pages 423–424, 1999. 17
- [13] N. Amenta and M. Bern. Surface Reconstruction by Voronoi Filtering. In *Proc. Symposium on Computational Geometry*, pages 39–48, 1998. 12, 17, 43, 46, 53, 99, 100, 103
- [14] N. Amenta, S. Choi, T. K. Dey, and N. Leekha. A Simple Algorithm for Homeomorphic Surface Reconstruction. In *Proc. Symposium on Computational Geometry*, pages 213–222, 2000. 17, 29, 101
- [15] N. Amenta, S. Choi, and R. Kolluri. The power crust. In *Proc. Symposium on Solid Modeling and Applications*, pages 249–266, 2001. 16, 56
- [16] N. Amenta, S. Choi, and R. Kolluri. The power crust, unions of balls, and the medial axis transform. volume 19, pages 127–153, 2001. 99
- [17] N. Amenta and Y. Kil. Defining point-set surfaces. In *ACM Transactions on Graphics (Proc. SIGGRAPH)*, pages 264–270, 2004. 21

- [18] M. Andersson, J. Giesen, M. Pauly, and B. Speckman. Bounds on the k-Neighborhood for Locally Uniform Sampled Surfaces. In *Proc. Symposium on Point-Based Graphics*, pages 167–171, 2004. [104](#)
- [19] A. Angelidis and M.-P. Cani. Adaptive Implicit Modeling using Subdivision Curves and Surfaces as Skeletons. In *Proc. Solid Modeling and Applications*, pages 45–52, 2002. [112](#), [155](#)
- [20] F. Arman and J. K. Aggarwal. Model-based object recognition in dense range images – a review. *ACM Computing surveys*, 25(1) :5–43, 1993. [3](#)
- [21] D. Attali, J.-D. Boissonnat, and H. Edelsbrunner. Stability and computation of medial axes : a state of the art report. In T. Möller, B. Hamann, and B. Russell, editors, *Mathematical Foundations of Scientific Visualization, Computer Graphics, and Massive Data Exploration*. Springer-Verlag, Mathematics and Visualization, 2007. [43](#), [99](#)
- [22] D. Attali, J.-D. Boissonnat, and A. Lieutier. Complexity of the Delaunay triangulation of Points on Surfaces : the Smooth Case. In *Proc. Symposium on Computational Geometry*, pages 201–210, 2003. [98](#)
- [23] M. Attene and M. Spagnuolo. Automatic surface reconstruction from point sets in space. In *Proc. Eurographics*, pages 457–465, 2000. [13](#)
- [24] C. Bajaj, J. Blinn, J. Bloomenthal, M.-P. Cani-Gascuel, A. Rockwood, B. Wyvill, and G. Wyvill. *Introduction to Implicit Surfaces*. Morgan-Kaufmann, 1997. [7](#)
- [25] C. B. Barber, D. P. Dobkin, and H. Huhdanpaa. The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software*, 22(4) :469–483, 1996. [56](#)
- [26] A. Barbier and E. Galin. Fast distance computation between a point and cylinders, cones, line swept spheres and cone-spheres. *Journal of Graphics Tools*, 9(2) :31–39, 2004. [111](#), [128](#)
- [27] A. Barbier, E. Galin, and S. Akkouche. Complex Skeletal Implicit Surfaces with Levels of Detail. *Journal of WSCG*, 12(1) :35–42, 2004. [111](#), [128](#), [129](#), [144](#)
- [28] A. H. Barr. Superquadrics and angle-preserving transforms. *IEEE Computer Graphics and Applications*, 1(1) :11–23, 1981. [108](#)
- [29] A. H. Barr. Global and Local Deformations of Solid Primitives. *Proc. SIGGRAPH*, 18(3) :21–30, 1984. [110](#), [113](#), [136](#), [137](#)
- [30] L. Barthe, V. Gaildrat, and R. Caubet. Extrusion of 1D profiles : Theory and first application. *International Journal of Shape Modeling*, 7(2) :179–198, 2001. [110](#)
- [31] B. G. Baumgart. Winged edge polyhedron representation. Technical Report AIM-179 (CS-TR-74-320), Computer Science Department, Stanford University, Palo Alto, CA, October 1972. [113](#)
- [32] D. Bechmann. Space deformation models survey. *Computer & Graphics*, 18(4) :571–586. [113](#)
- [33] F. Bernardini, J. Mittleman, H. Rushmeier, C. Silva, and G. Taubin. The Ball-Pivoting Algorithm for Surface Reconstruction. *IEEE Transactions on Visualization and Computer Graphics*, 5(4) :349–359, 1999. [14](#)
- [34] C. Blanc. Generic implementation of axial deformation techniques. pages 139–145, 1995. [110](#)
- [35] C. Blanc and C. Schlick. Extended field functions for soft objects. In *Proc. Implicit Surfaces*, pages 21–32, 1995. [111](#)
- [36] J. F. Blinn. A generalization of algebraic surface drawing. *ACM Transactions on Graphics*, 1(3) :235–256, 1982. [108](#), [110](#), [111](#), [133](#)
- [37] J. Bloomenthal and K. Shoemake. Convolution surfaces. *Computer Graphics (Proc. SIGGRAPH)*, 25(4) :251–256, 1991. [112](#)

-
- [38] J.-D. Boissonnat. Geometric structures for three-dimensional shape representation. *ACM Transactions on Graphics*, 3(4) :226–286, 1984. [12](#), [14](#)
- [39] J.-D. Boissonnat and F. Cazals. Smooth surface reconstruction via natural neighbor interpolation of distance functions. In *Proc. Symposium on Computational Geometry*, pages 223–232, 2000. [16](#), [20](#), [30](#), [32](#), [99](#)
- [40] J.-D. Boissonnat and F. Cazals. Coarse-to-fine surface simplification with geometric guarantees. In *Proc. Eurographics*, pages 490–499, 2001. [17](#), [30](#), [33](#)
- [41] J.-D. Boissonnat, D. Cohen-Steiner, and G. Vegter. Isotopic Implicit Surface Meshing. In *Proc. ACM Symposium on Theory of Computing*, pages 301–309, 2004. [120](#)
- [42] J.-D. Boissonnat and S. Oudot. Provably Good Surface Sampling and Approximation. In *Proc. Symposium on Geometry Processing*, 2003. [31](#), [120](#)
- [43] J.-D. Boissonnat and S. Oudot. An effective Condition for Sampling Surfaces with Guarantees. In *Proc. Symposium on Solid Modeling and Applications*, pages 101–112, 2004. [21](#), [31](#), [120](#)
- [44] J.-D. Boissonnat and S. Oudot. Provably good sampling and meshing of surfaces. *Graphical Models*, 67(5) :405–451, 2005. [31](#), [120](#)
- [45] J.-D. Boissonnat and S. Oudot. Provably good sampling and meshing of lipschitz surfaces. In *Proc. Symposium on Computational Geometry*, pages 337–346, 2006. [46](#), [120](#)
- [46] J.-D. Boissonnat and M. Yvinec. *Géométrie Algorithmique*. Ediscience international, Paris, 1995. [97](#)
- [47] P. Borrel and D. Bechmann. Deformation of n-dimensional objects. In *Proc. Solid Modeling and Applications*, pages 351–369, 1991. [113](#), [137](#)
- [48] S. Campagna, L. Kobbelt, and H.-P. Seidel. Directed Edges - A Scalable Representation for Triangle Meshes. *Journal of Graphics Tools*, 3(4) :1–12, 1998. [113](#), [149](#)
- [49] M.-P. Cani and M. Desbrun. Animation of Deformable Models using Implicit Surfaces. *IEEE Transactions on Visualization and Computer Graphics*, 3(1) :39–50, 1997. [110](#), [133](#)
- [50] J. C. Carr, R. K. Beatson, J. B. Cherrie, T. J. Mitchell, W. R. Froght, B. C. McCalum, and T. R. Evans. Reconstruction and representation of 3D objects with radial basis functions. In *Proc. of SIGGRAPH 2001*, pages 67–76, 2001. [22](#)
- [51] F. Cazals and J. Giesen. Delaunay Triangulation based Surface Reconstruction : Ideas and Algorithms. Technical Report 5393, INRIA, November 2004. [4](#), [11](#), [97](#)
- [52] R. Chaine. A geometric convection approach of 3-D reconstruction. In *Proc. Symposium on Geometry Processing*, pages 218–229, 2003. [6](#), [13](#), [35](#), [39](#), [40](#)
- [53] F. Chazal and A. Lieutier. The lambda medial axis. *Graphical Models*, 67(4) :304–331, 2005. [23](#), [43](#), [99](#)
- [54] B. Chazelle. An optimal algorithm for intersecting three-dimensional convex polyhedra. *SIAM J. Comput.*, 21(4) :671–696, 1992. [114](#)
- [55] Y. Chen and G. Medioni. Object modeling by registration of multiple range images. *Proc. IEEE International Conference on Robotics and Automation*, 3 :2724–2729, 1991. [3](#)
- [56] E. V. Chernyaev. Marching Cubes 33 : Construction of topologically correct isosurfaces. Technical Report CN/95-17, CERN, March 1995. [119](#)
- [57] L. P. Chew. Guaranteed-Quality Mesh Generation for Curved Surfaces. In *Proc. of Symposium on Computational Geometry*, pages 274–280, 1993. [31](#), [100](#), [120](#)

- [58] P. Cignoni, F. Ganovelli, C. Montani, and R. Scopigno. Reconstruction of topologically correct and adaptive trilinear surfaces. *Computers & Graphics*, 24(3) :399–418, 2000. [119](#)
- [59] P. Cignoni, F. Ganovelli, E. Gobbetti, F. Marton, F. Ponchio, and R. Scopigno. Adaptive Tetra-Puzzles – Efficient Out-of-core Construction and Visualization of Gigantic Polygonal Models. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 23(3) :796–803, 2004. [85](#)
- [60] P. Cignoni, C. Montani, C. Rocchini, and R. Scopigno. IEEE Transactions on Visualization and Computer Graphics. *External Memory Management and Simplification of Huge Meshes*, 9(4) :525–537, 2003. [85](#)
- [61] D. Cohen-Steiner, P. Alliez, and M. Desbrun. Variational Shape Approximation. In *ACM Transactions on Graphics (Proc. SIGGRAPH)*, pages 905–914, 2004. [4](#)
- [62] D. Cohen-Steiner and F. Da. A greedy delaunay-based surface reconstruction algorithm. *The Visual Computer*, 20(1) :4–16, 2004. [14](#)
- [63] S. Coquillart. Extended Free-Form Deformation : A Sculpturing Tool for 3D Geometric Modeling. In *Proc. SIGGRAPH*, pages 187–196, 1990. [113](#), [137](#)
- [64] B. Crespín. Implicit Free-Form Deformations. In *Proc. Implicit Surfaces*, pages 17–23, 1999. [121](#)
- [65] B. Crespín, P. Guitton, and C. Schlick. Efficient and accurate tessellation of implicit sweeps. In *CSG’98 (Set-theoretic Solid Modeling Technics and Applications)*, pages 49–63, April 1998. [120](#)
- [66] B. Crespín, C. Blanc, and C. Schlick. Implicit sweep objects. In *Proc. Eurographics*, pages 165–174, 1996. [111](#)
- [67] B. Curless and M. Levoy. A Volumetric Method for Building Complex Models from Range Images. *Computer Graphics (Proc. SIGGRAPH)*, 30 :303–312, 1996. [23](#), [64](#), [104](#)
- [68] J. Davis, S. R. Marschner, M. Garr, and M. Levoy. Filling holes in complex surfaces using volumetric diffusion. In *First International Symposium on 3D Data Processing, Visualization, and Transmission*, June 19-21, 2002. [4](#)
- [69] P. Decaudin. Geometric Deformation by Merging a 3D-Object with a Simple Shape. In *Proc. Graphics Interface*, pages 55–60, 1996. [121](#)
- [70] M. Desbrun, N. Tsingos, and M.-P. Cani. Adaptive sampling of implicit surfaces for interactive modeling and animation. *Computer Graphics Forum*, 15(5), 1996. Published under the name M.-P. Gascuel. [120](#)
- [71] O. Devillers. The Delaunay hierarchy. *International Journal of Foundations of Computer Science*, 13(2) :163–180, 2002. [98](#)
- [72] T. K. Dey. Curve and surface reconstruction. *Chapter in Handbook of Discrete and Computational Geometry, Goodman and O’ Rourke eds., CRC press, 2nd edition*, 2004. [11](#)
- [73] T. K. Dey, S. Funke, and E. A. Ramos. Surface Reconstruction in almost Linear Time under Locally Uniform Sampling. In *Proc. European Workshop on Computational Geometry*, 2001. [18](#)
- [74] T. K. Dey and J. Giesen. Detecting undersampling in surface reconstruction. In *Proc. Symposium on Computational Geometry*, pages 257–263, 2001. [18](#)
- [75] T. K. Dey, J. Giesen, S. Goswami, J. Hudson, R. Wenger, and W. Zhao. Undersampling and oversampling in sample based shape modeling. In *Proc. IEEE Visualization*, pages 83–90, 2001. [18](#)
- [76] T. K. Dey, J. Giesen, S. Goswami, and W. Zhao. Shape Dimension and Approximation from Samples. In *Proc. ACM-SIAM Symposium on Discrete Algorithms*, pages 772–780, 2002. [100](#)

-
- [77] T. K. Dey, J. Giesen, and J. Hudson. Decimating samples for mesh simplification. In *Proc. Canadian Conference on Computational Geometry*, pages 85–88, 2001. [18](#), [29](#)
- [78] T. K. Dey, J. Giesen, and J. Hudson. Delaunay-based shape reconstruction from large data. In *Proc. IEEE Symposium in Parallel and Large Data Visualization and Graphics*, pages 19–27, 2001. [18](#), [85](#)
- [79] T. K. Dey, J. Giesen, and J. Hudson. Sample shuffling for quality hierarchic surface meshing. In *Proc. 10th International Meshing Roundtable Conference*, pages 143–154, 2001. [18](#), [29](#), [30](#)
- [80] T. K. Dey, J. Giesen, E. A. Ramos, and B. Sadri. Critical points of the distance to an epsilon-sampling of a surface and flow-complex-based surface reconstruction. In *Symposium on Computational Geometry*, pages 218–227, 2005. [13](#), [40](#)
- [81] T. K. Dey and S. Goswami. Tight cocone : A water-tight surface reconstructor. *3* :302–307, 2003. [18](#)
- [82] T. K. Dey and S. Goswami. Provable surface reconstruction from noisy samples. In *Proc. Symposium on Computational Geometry*, pages 330–339, 2004. [16](#), [18](#)
- [83] T. K. Dey and J. Sun. An adaptive MLS surface for reconstruction with guarantees. In *Proc. Symposium on Geometry Processing*, pages 43–52, 2005. [21](#)
- [84] K. Dobrindt, K. Mehlhorn, and M. Yvinec. A complete and efficient algorithm for the intersection of a general ans a convex polyhedron. Technical Report 2023, INRIA, September 1993. [114](#)
- [85] H. Edelsbrunner. Surface reconstruction by wrapping finite point sets in space. In B. Aronov, S. Basu, J. Pach, and S.-V. M. Sharir, editors, *Discrete and Computational Geometry, The Pollack-Goodman Festschrift*, pages 379–404. Springer-Verlag, 2002. [13](#), [38](#)
- [86] H. Edelsbrunner, D. G. Kirkpatrick, and R. Seidel. On the shape of a set of points in the plane. *IEEE Transactions on Information Theory*, *29* :51–559, 1983. [12](#)
- [87] H. Edelsbrunner and E. P. Mücke. Simulation of simplicity : A technique to cope with degenerate cases in geometric algorithms. *ACM Transactions on Graphics*, *9*(1) :66–104, 1990. [98](#)
- [88] H. Edelsbrunner and E. P. Mücke. Three-dimensional alpha shapes. *ACM Transactions on Graphics*, *13*(1) :43–72, 1994. [14](#)
- [89] H. Edelsbrunner and N. R. Shah. Triangulating Topological Spaces. In *Proc. Symposium on Computational Geometry*, pages 285–292, 1994. [12](#), [100](#)
- [90] E. Ferley, M.-P. Cani, and J.-D. Gascuel. Practical volumetric sculpting. 1999. [112](#)
- [91] S. Fleishman, C. T. Silva, and D. Cohen-Or. Robust Moving Least-squares Fitting with Sharp Features. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, *24*(3) :544–552, 2005. [21](#)
- [92] M. Fox, C. Galbraith, and B. Wyvill. Efficient Implementation of the Blobtree for Rendering Purposes. In *Proc. Shape Modeling International*, pages 306–314, 2001. [138](#), [144](#)
- [93] S. F. Frisken, R. N. Perry, A. P. Rockwood, and T. R. Jones. Adaptively Sampled Distance Fields : A General Representation of Shape for Computer Graphics. In *Proc. SIGGRAPH*, pages 249–254, 2000. [112](#), [153](#)
- [94] S. Funke and E. A. Ramos. Smooth-Surface Reconstruction in Near-Linear Time. In *Proc. ACM-SIAM Symposium on Discrete Algorithms*, pages 781–790, 2002. [18](#), [100](#)
- [95] T. Funkhouser, M. Kazhdan, P. Shilane, P. Min, W. Kiefer, A. Tal, S. Rusinkiewicz, and D. Dobkin. Modeling by Example. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, *23*(3) :652–663, 2004. [114](#)

- [96] E. Galin and S. Akkouche. Fast Processing of Triangle Meshes using Triangle Fans. In *Proc. Shape Modeling International*, pages 326–331, 2005. 132
- [97] Y. Gardan and E. Perrin. An algorithm reducing 3D Boolean operations to a 2D problem : concepts and results. *Computer-Aided Design*, 28(4) :277–287, 1996. 114
- [98] M. Garland. *Quadric-Based Polygonal Surface Simplification*. PhD thesis, Computer Science Department, Carnegie Mellon University, May 1999. 26
- [99] M. Garland and P. S. Heckbert. Surface Simplification using Quadric Error Metrics. In *Proc. SIGGRAPH*, pages 209–216, 1997. 4, 29
- [100] N. Gelfand, N. Mitra, L. Guibas, and H. Pottmann. Robust global registration. In *Proc. Symposium on Geometry Processing*, pages 197–206, 2005. 3
- [101] J. Giesen and M. John. The Flow Complex : A Data Structure for Geometric Modeling. In *Proc. ACM-SIAM Symposium on Discrete Algorithms*, pages 285–294, 2003. 13, 38
- [102] G. Godin, J.-A. Beraldin, M. Rioux, M. Levoy, and L. Cournoyer. An assessment of laser range measurement of marble surfaces. In *Proc. Conference on Optical 3D Measurement Techniques*, pages 49–56, 2001. 3
- [103] G. Godin, J.-A. Beraldin, J. Taylor, L. Cournoyer, M. Rioux, S. El-Hakim, R. Baribeau, F. Blais, P. Boulanger, J. Dorney, and M. Picard. Active optical 3d imaging for heritage applications. *IEEE Computer Graphics and Applications*, 22(5) :24–36, 2002. 3
- [104] M. Gopi, S. Krishnan, and C. T. Silva. Surface reconstruction based on lower dimensional localized delaunay triangulation. In *Proc. Eurographics*, pages 467–478, 2000. 15
- [105] A. Guézic. Locally toleranced surface simplification. *IEEE Transactions on Visualization and Computer Graphics*, 5(2) :168–189, 1999. 68
- [106] P. Guigue and O. Devillers. Fast and Robust Triangle-Triangle Overlap Test Using Orientation Predicates. *Journal of Graphics Tools*, 8(1) :25–32, 2003. 148
- [107] A. Guézic. Meshsweeper : Dynamic Point-to-Polygonal-Mesh Distance and Applications. *IEEE Transactions on Visualization and Computer Graphics*, 7(1) :47–61, 2001. 118
- [108] P. Hachenberger and L. Kettner. Boolean operations on 3d selective nef complexes : Optimized implementation and experiments. In *Proc. ACM Symposium on Solid and Physical Modeling*, pages 163–174, 2005. 114
- [109] A. Hilton, A. J. Stoddart, J. Illingworth, and T. Windeatt. Marching Triangles : Range image fusion for complex object modelling. In *IEEE International Conference on Image Processing*, pages 381–384, 1996. 119
- [110] C.-C. Ho, F.-C. Wu, B.-Y. Chen, Y.-Y. Chuang, and M. Ouhyoung. Cubical marching squares : Adaptive feature preserving surface extraction from volume data. *Computer Graphics Forum (Proc. Eurographics)*, 24(3) :537–545, 2005. 119
- [111] H. Hoppe. Progressive Meshes. In *Proc. SIGGRAPH*, pages 99–108, 1996. 148
- [112] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Surface Reconstruction from Unorganized Points. *Computer Graphics (Proc. SIGGRAPH)*, 26(2) :71–78, 1992. 19, 104
- [113] A. Hornung and L. Kobbelt. Robust reconstruction of watertight 3d models from non-uniformly sampled point clouds without normal information. In *Proc. Symposium on Geometry Processing*, 2006. To appear. 24, 32
- [114] G. Impoco, P. Cignoni, and R. Scopigno. Closing gaps by clustering unseen directions. pages 307–316, 2004. 3

-
- [115] M. Isenburg, Y. Liu, J. Shewchuk, and J. Snoeyink. Streaming Computation of Delaunay Triangulations. In *Proc. SIGGRAPH*, 2006. To appear. [94](#)
- [116] D. Johnson and E. Cohen. A Framework for Efficient Minimum Distance Computation. In *Proc. Conf. Robotics and Automation*, pages 3678–3683, 1998. [131](#)
- [117] T. Ju, F. Losasso, S. Schaefer, and J. Warren. Dual contouring of hermite data. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 21(3) :339–346, 2002. [119](#)
- [118] T. Kanai, H. Suzuki, J. Mintani, and F. Kimura. Interactive Mesh Fusion Based on Local 3D Metamorphosis. In *Proc. Graphics Interface '99*, pages 148–156, 1999. [114](#), [153](#)
- [119] T. Karkanis and A. J. Stewart. High Quality, Curvature Dependent Triangulation of Implicit Surfaces. *IEEE Computer Graphics and Applications*, 21(2) :60–69, March 2001. [119](#)
- [120] M. Kazhdan. Reconstruction of solid models from oriented point sets. In *Proc. Symposium on Geometry Processing*, pages 73–82, 2005. [23](#)
- [121] L. P. Kobbelt and M. Botsch. Freeform Shape Representations for Efficient Geometry Processing. In *Proc. Shape Modeling International*, pages 111–115, 2003. [120](#)
- [122] L. P. Kobbelt, M. Botsch, U. Schwanecke, and H.-P. Seidel. Feature Sensitive Surface Extraction from Volume Data. In *Proc. SIGGRAPH*, pages 57–66, 2001. [119](#)
- [123] R. Kolluri. Provably good moving least squares. In *Proc. ACM-SIAM Symposium on Discrete Algorithms*, pages 1008–1017, 2005. [21](#)
- [124] R. Kolluri, J. R. Shewchuk, and J. F. O’Brien. Spectral surface reconstruction from noisy point clouds. In *Proc. Symposium on Geometry Processing*, pages 11–21, 2004. [16](#)
- [125] D. Levin. The approximation power of moving least-squares. *Math. Comp.*, 67(224) :1517–1531, 1998. [21](#)
- [126] D. Levin. Mesh-independent surface interpolation. *Geometric Modeling for Scientific Visualization*, 2003. [21](#), [32](#)
- [127] M. Levoy, K. Pulli, B. Curless, S. Rusinkiewicz, D. Koller, L. Pereira, M. Ginzton, S. Anderson, J. Davis, J. Ginsberg, J. Shade, and D. Fulk. The Digital Michelangelo Project : 3D Scanning of Large Statues. In *Proc. SIGGRAPH*, pages 131–144, 2000. [1](#), [3](#), [23](#)
- [128] T. Lewiner, H. Lopes, A. W. Vieira, and G. Tavares. Efficient Implementation of Marching Cubes’ Cases with Topological Guarantees. *Journal of Graphics Tools*, 8(2) :1–15, 2003. [119](#)
- [129] P. Lindstrom. Out-of-core simplification of large polygonal models. In *Proc. SIGGRAPH*, pages 259–262, 2000. [85](#)
- [130] L. Linsen. Point cloud representation. Technical Report 2001-3, Universität Karlsruhe, Germany, 2001. [29](#), [103](#)
- [131] P. Linstrom and C. T. Silva. A memory insensitive technique for large model simplification. In *Proc. IEEE Visualization*, pages 121–126, 2001. [4](#)
- [132] P. Linstrom and G. Turk. Fast and memory efficient polygonal simplification. In *Proc. IEEE Visualization*, pages 279–286, 1998. [4](#)
- [133] W. E. Lorensen and H. E. Cline. Marching Cubes : A high Resolution 3D surface reconstruction algorithm. *Computer Graphics (Proc. SIGGRAPH)*, 21(4) :163–169, 1987. [23](#), [119](#), [140](#), [141](#)
- [134] R. MacCracken and K. I. Joy. Free-form Deformations with Lattices of Arbitrary Topology. In *Proc. SIGGRAPH*, pages 181–188, 1996. [113](#), [137](#)
- [135] M. Mäntylä. *An introduction to solid modeling*. Computer Science Press, Rockville, Md., 1988. [113](#)

- [136] A. Martinet, E. Galin, B. Desbenoit, and S. Akkouche. Procedural Modeling of Cracks and Fractures. In *Proc. Shape Modeling International*, pages 346–349, 2004. [152](#)
- [137] S. Mauch. A fast algorithm for computing the closest point and distance transform. Technical report, Cal. Institute of Technology, 2000. [118](#)
- [138] B. Mederos, N. Amenta, L. Velho, and L. H. de Figueiredo. Surface reconstruction from noisy point clouds. In *Proc. Symposium on Geometry Processing*, pages 53–62, 2005. [16](#)
- [139] B. Mederos, L. Velho, and L. H. D. Figueiredo. Moving least squared multiresolution surface approximation. In *Proc. SIBGRAPI - Brazilian Symposim on Computer Graphics and Image Processing*, pages 19–26, 2003. [21](#)
- [140] R. Mencl and H. Müller. Interpolation and approximation of surfaces from three-dimensional scattered data points. *Eurographics State of the Art Report*, pages 51–67, 1998. [4](#), [11](#)
- [141] K.-H. Min, I.-K. Lee, and C.-M. Park. Component Based Polygonal Approximation of Soft Objects. *Computers & Graphics*, 25 :245–257, 2001. [120](#)
- [142] N. J. Mitra, N. Gelfand, H. Pottmann, and L. Guibas. Registration of point cloud data from a geometric optimization perspective. In *Symposium on Geometry Processing*, pages 23–32, 2004. [3](#)
- [143] N. J. Mitra and A. Nguyen. Estimating surface normals in noisy point cloud data. In *Proc. Symposium on Computational Geometry*, pages 322–328, 2003. [103](#)
- [144] C. Moenning and N. A. Dogson. A new point cloud simplification algorithm. In *Proc. Int. Conf. on Visualization, Imaging and Image Processing*, pages 1027–1033, 2003. [31](#)
- [145] C. Moenning and N. A. Dogson. Intrinsic point cloud simplification. In *Proc. GraphiCon*, 2004. [31](#)
- [146] T. A. Möller. Fast 3D Triangle-Box Overlap Testing. *Journal of Graphics Tools*, 6(1) :29–33, 2001. [63](#)
- [147] B. S. Morse, T. S. Yoo, D. T. Chen, P. Rheingans, and K. R. Subramanian. Interpolating implicit surfaces from scattered surface data using compactly supported radial basis functions. In *Proc. Shape Modeling International*, pages 89–98. IEEE Computer Society Press, 2001. [22](#)
- [148] K. Museth, D. E. Breen, R. T. Whitacker, and A. H. Barr. Level Set Surface Editing Operators. *ACM Transactions on Graphics*, 21(3) :330–338, 2002. [112](#), [153](#)
- [149] B. K. Natarajan. On generating topologically consistent isosurfaces from uniform samples. *The Visual Computer*, 11(1) :52–62, 1994. [119](#)
- [150] T. Nishimura, M. Hirai, T. Kawai, T. Kawata, I. Shirakawa, and K. Omura. Object modeling by distribution function and a method of image generation. In *Proc. Electronics Communication Conference*, 1985. [110](#)
- [151] Y. Ohtake and A. Belyaev. Dual/Primal optimization for polygonized implicit surfaces. In *Proc. of Solid Modelling 2002*, Saarbrücken, Germany, 2002. [120](#)
- [152] Y. Ohtake, A. Belyaev, M. Alexa, G. Turk, and H.-P. Seidel. Multi-level Partition of Unity Implicit. *ACM Transactions on Graphics*, 22(3) :463–470, 2003. [20](#), [32](#)
- [153] Y. Ohtake, A. Belyaev, and A. Pasko. Dynamic mesh optimization for polygonized implicit surfaces with sharp features. *The Visual Computer*, 19(2), 2003. [120](#)
- [154] Y. Ohtake, A. Belyaev, and H.-P. Seidel. A multi-scale approach to 3D scattered data interpolation with compactly supported basis functions. In *Proc. Shape Modeling International*, pages 153–161, 2003. [22](#)

-
- [155] Y. Ohtake, A. G. Belyaev, and H.-P. Seidel. 3D scattered data approximation with adaptive compactly supported radial basis functions. In *Proc. Shape Modeling International*, pages 31–39. IEEE Computer Society Press, 2004. [23](#)
- [156] Y. Ohtake, A. G. Belyaev, and H.-P. Seidel. An integrating approach to meshing scattered point data. In *Proc. Symposium on Solid and Physical Modeling*, pages 61–69, 2005. [15](#), [28](#), [30](#), [33](#), [70](#), [94](#)
- [157] A. Pasko, V. Adzhizev, A. Sourin, and V. Savchenko. Function Representation in Geometric Modeling : Concepts, Implementation and Applications. *The Visual Computer*, 11(8) :429–446, 1995. [109](#), [110](#), [135](#)
- [158] G. Pasko, A. Pasko, M. Ikeda, and T. Kunii. Bounded Blending Operations. In *Proc. Shape Modeling International*, pages 95–104, 2002. [110](#), [133](#)
- [159] M. Pauly and M. Gross. Spectral processing of point-sampled geometry. In *Computer Graphics (Proc. SIGGRAPH)*, pages 379–386, 2001. [31](#)
- [160] M. Pauly, M. Gross, and L. P. Kobbelt. Efficient Simplification of Point-Sampled Surfaces. In *Proc. IEEE Visualization Conference*, pages 163–170, 2002. [27](#), [29](#), [31](#)
- [161] M. Pauly, R. Keiser, L. P. Kobbelt, and M. Gross. Shape Modeling with Point-Sampled Geometry. *ACM Transactions on Graphics*, 22(3) :641–650, 2003. [122](#)
- [162] P. Reuter, I. Tobor, C. Schlick, and S. Dedieu. Point-based Modelling and Rendering using Radial Basis Functions. In *Proc. ACM Graphite 2003*, pages 111–118, 2003. [122](#)
- [163] A. Ricci. A constructive geometry for computer graphics. *The Computer Journal*, 16(2) :157–160, 1973. [109](#), [135](#)
- [164] C. Rocchini, P. Cignoni, F. Ganovelli, C. Montani, P. Pingi, , and R. Scopigno. The marching intersections algorithm for merging range images. *The Visual Computer*, 20(2–3) :149–164, May 2004. [23](#), [64](#), [104](#)
- [165] M. Samozino, P. Alliez, and M. Yvinec. Reconstruction with voronoi centered radial basis functions. In *Proc. Symposium on Geometry Processing*, pages 51–60, 2006. [22](#), [32](#)
- [166] M. Sarifuddin and R. Missaoui. A new perceptually uniform color space with associated color similarity measure for content-based image and video retrieval. In *Proc. ACM SIGIR Workshop on Multimedia Information Retrieval*, 2005. [65](#)
- [167] V. Savchenko, A. Pasko, O. G. Okunev, and T. L. Kunii. Function representation of solids reconstructed from scattered surface points and contours. *Computer Graphics Forum*, 14(4) :181–188, 1995. [22](#)
- [168] O. Schall, A. Belyaev, and H.-P. Seidel. Robust filtering of noisy scattered point data. In *Proc. Point-Based Graphics 2005*, pages 71–77, 2005. [4](#)
- [169] O. Schall and M. Samozino. Surface from scattered points : A brief survey of recent developments. In *First International Workshop on Semantic Virtual Environments*, pages 138–147, 2005. [4](#), [11](#)
- [170] C. E. Scheidegger, S. Fleishman, and C. T. Silva. Triangulating point set surfaces with bounded error. In *Proc. Symposium on Geometry Processing*, pages 63–72, 2005. [21](#)
- [171] R. Schmidt, B. Wyvill, and E. Galin. Interactive Implicit Modeling With Hierarchical Spatial Caching. In *Proc. Shape Modeling International*, pages 104–113, 2005. [153](#)
- [172] B. Schmitt, A. Pasko, and C. Schlick. Shape-Driven Deformations of Functionally Defined Heterogeneous Volumetric Objects. In *Proc. ACM Graphite 2003*, pages 127–134, 2003. [121](#)

- [173] P. Schneider and D. H. Eberly. *Geometric Tools for Computer Graphics*. Morgan Kaufman Series in Computer Graphics and Geometric Modeling, 2002. [111](#), [128](#)
- [174] T. W. Sederberg and S. R. Parry. Free-Form Deformation of Solid Geometric Models. In *Proc. SIGGRAPH*, pages 151–160, 1986. [7](#), [110](#), [113](#), [136](#), [137](#)
- [175] A. Sharf, T. Lewiner, A. Shamir, L. Kobbelt, and D. Cohen-Or. Competing fronts for coarse-to-fine surface reconstruction. In *Proc. Eurographics*, 2006. To appear. [26](#)
- [176] C. Shen, J. F. O’Brien, and J. R. Shewchuk. Interpolating and Approximating Implicit Surfaces from Polygon Soup. *ACM Transactions on Graphics*, 23(3) :896–904, 2004. [118](#)
- [177] K. Singh and R. Parent. Polyhedral shapes as general implicit surface primitives. Technical Report OSU-CISRC-5/94-TR24, Ohio State University CIS, 1994. [111](#)
- [178] K. Singh and R. Parent. Joining Polyhedral Objects using Implicitly Defined Surfaces. *The Visual Computer*, 17(7) :415–428, 2001. [121](#), [153](#)
- [179] O. Sorkine, D. Cohen-Or, Y. Lipman, M. Alexa, C. Rössl, and H.-P. Seidel. Laplacian Surface Editing. In *Proc. Eurographics/ACM SIGGRAPH Symposium on Geometry Processing*, pages 179–188, 2004. [115](#)
- [180] H. Theisel. Exact isosurfaces for marching cubes. *Computer Graphics Forum*, 21(1) :19–31, 2002. [119](#)
- [181] I. Tobor, P. Reuter, and C. Schlick. Efficient reconstruction of large scattered geometric datasets using the partition of unity and radial basis functions. *Journal of WSCG*, 12(3) :467–474, 2004. [20](#), [23](#)
- [182] G. Turk and J. F. O’Brien. Variational implicit surfaces. Technical Report GIT-GVU-99-15, Graphics, Visualization, and Useability Center, Georgia Institute of Technology, 1999. [22](#)
- [183] G. Turk and J. F. O’Brien. Modelling with implicit surfaces that interpolate. *ACM Transactions on Graphics*, 21(4), 2002. [22](#)
- [184] C. W. A. M. van Overveld and B. Wyvill. Shrinkwrap : an adaptive algorithm for polygonizing an implicit surface. Technical Report 93/514/19, University of Calgary, March 1993. [120](#)
- [185] L. Velho. Simple and efficient polygonization of implicit surfaces. *Journal of Graphics Tools*, 1(2) :5–24, 1996. [119](#), [120](#)
- [186] L. Velho, J. Gomes, and L. H. Figueiredo. *Implicit Objects in Computer Graphics*. Springer Verlag, New York, 2002. [7](#)
- [187] R. C. Veltkamp. Boundaries through scattered points of unknown density. *Graphical Models and Image Processing*, 57(20) :441–452, 1995. [13](#)
- [188] T. Weyrich, M. Pauly, R. Keiser, S. Heinzle, S. Scandella, and M. Gross. Post-processing of scanned 3d surface data. In *Symposium on Point-Based Graphics*, 2004. [4](#)
- [189] A. P. Witkin and P. S. Heckbert. Using particles to sample and control implicit surfaces. *Computer Graphics*, 28(2) :269–277, 1994. [31](#), [119](#)
- [190] J. Wu and L. P. Kobbelt. Optimized Sub-Sampling of Point Sets for Surface Splatting. In *Proc. Eurographics*, pages 643–652, 2004. [27](#), [28](#)
- [191] B. Wyvill, E. Galin, and A. Guy. Extending The CSG Tree. Warping, Blending and Boolean Operations in an Implicit Surface Modeling System. *Computer Graphics Forum*, 18(2) :149–158, June 1999. [8](#), [109](#), [111](#), [138](#)
- [192] B. Wyvill, P. Jepp, K. van Overveld, and G. Wyvill. Subdivision surfaces for fast approximate implicit polygonization. Technical Report 2000-671-23, University of Calgary, Dept. of Computer Science, 2000. [120](#)

-
- [193] B. Wyvill and G. Wyvill. Field functions for implicit surfaces. *The Visual Computer*, 5 :75–82. [108](#)
- [194] G. Wyvill, C. McPheeters, and B. Wyvill. Data Structure for Soft Objects. *The Visual Computer*, 2(4) :227–234, 1986. [110](#), [119](#)
- [195] H. Xie, J. Wang, J. Hua, H. Qin, and A. Kaufman. Piecewise C^1 continuous surface reconstruction of noisy point clouds via local implicit quadric regression. In *Proc. IEEE Visualization*, pages 91–98, 2003. [21](#)
- [196] D. Xu, H. Zhang, Q. Wang, and H. Bao. Poisson Shape Interpolation. In *Proc. ACM Symposium on Solid and Physical Modeling*, pages 267–274, 2005. [114](#)
- [197] Y. Yu, K. Zhou, D. Xu, X. Shi, H. Bao, B. Guo, and H.-Y. Shum. Mesh Editing with Poisson-based Gradient Field Manipulation. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 23(3) :644–651, 2004. [115](#)
- [198] H.-K. Zhao, S. Osher, and R. Fedkiw. Fast Surface Reconstruction using the Level Set Method. In *Proc. IEEE Workshop on Variational and Level Set Methods in Computer Vision (VLSM)*, pages 194–202, 2001. [13](#), [25](#), [35](#), [36](#)

Publications de l'auteur

- [1] R. Allègre, A. Barbier, E. Galin, and S. Akkouche. A hybrid shape representation for free-form modeling. In *Proc. Shape Modeling International*, pages 7–18. IEEE Computer Society Press, 2004.
- [2] R. Allègre, E. Galin, and S. Akkouche. A Hybrid Shape Representation for Free-Form Modeling. Technical Report RR-LIRIS-2004-009, LIRIS CNRS - Université Lyon 1, France, January 2004.
- [3] R. Allègre, R. Chaine, and S. Akkouche. Convection-Driven Dynamic Surface Reconstruction. In *Proc. Shape Modeling International*, pages 33–42. IEEE Computer Society Press, 2005.
- [4] R. Allègre, R. Chaine, and S. Akkouche. Convection-Driven Dynamic Surface Reconstruction. Technical Report RR-LIRIS-2005-009, LIRIS CNRS - Université Lyon 1, France, January 2005.
- [5] R. Allègre, R. Chaine, and S. Akkouche. Reconstruction dynamique de surface par convection géométrique. In *Actes des 18ièmes journées de l'Association Française d'Informatique Graphique (AFIG), 3ièmes journées de l'Association Chapitre Français d'Eurographics*, pages 182–192, 2005.
- [6] R. Allègre, E. Galin, R. Chaine, and S. Akkouche. The hybridtree : Mixing skeletal implicit surfaces, triangle meshes and point sets in a free-form modeling system. *Graphical Models*, 68(1) :42–64, 2006. SMI'04 special issue.
- [7] E. Galin, R. Allègre, and S. Akkouche. A Fast Particle System Framework for Interactive Implicit Modeling. In *Proc. Shape Modeling International*, pages 215–221. IEEE Computer Society Press, 2006.
- [8] R. Allègre, R. Chaine, and S. Akkouche. A Dynamic Surface Reconstruction Framework for Large Point Sets. In *Proc. IEEE/Eurographics Symposium on Point-Based Graphics*, pages 17–26, 2006.
- [9] R. Allègre, R. Chaine, and S. Akkouche. A Flexible Framework for Surface Reconstruction from Large Point Sets. *Computers & Graphics*, 31(2), 2007. PBG'06 special issue.

Titre Contributions à l'introduction de flexibilité dans la reconstruction et l'édition de modèles 3D.

Résumé Cette thèse s'inscrit dans la problématique générale de la modélisation géométrique de formes libres en trois dimensions. Dans le cadre du projet Art3D (ACI Masses de Données), nous avons exploré deux directions de recherche relatives à la représentation d'objets numérisés, avec un objectif de flexibilité : la reconstruction dynamique de surface et la modélisation multireprésentation. Dans la première partie de la thèse, nous présentons une approche combinatoire flexible de la reconstruction de surface permettant de produire directement une surface triangulée simplifiée à partir d'un échantillon de points dense. La surface reconstruite peut ensuite être mise à jour de manière dynamique en ajoutant ou en retirant des données localement. La seconde partie de la thèse est consacrée à l'étude d'un modèle géométrique hybride destiné à la création de formes complexes à partir d'objets numérisés ou de primitives simples. Nous avons développé un arbre de construction permettant de combiner de manière cohérente des surfaces implicites, des maillages polygonaux et des nuages de points.

Mots-clés Reconstruction dynamique de surface, convection géométrique, simplification d'échantillons de points, modèle hybride, surfaces implicites, maillages polygonaux, mélange, déformations locales.

Title Contributions to the introduction of flexibility in 3D model reconstruction and editing.

Abstract This thesis deals with geometric modeling of complex 3D free-form shapes. In the context of the Art3D project (ACI Masses de Données), we have investigated two research directions related to the representation of digitized objects, with a flexibility goal : dynamic surface reconstruction and multi-representation modeling. In the first part of the thesis, we introduce a flexible combinatorial approach to surface reconstruction that makes it possible to directly generate a simplified triangulated surface from a dense input point set. The reconstructed surface can be next locally refined or coarsened in a dynamic fashion. The second part presents a hybrid modeling framework for creating complex shapes from scanned models or simple primitives. We have developed an extended CSG tree that mixes implicit surfaces, polygonal meshes, and point set models in a coherent fashion.

Keywords Dynamic surface reconstruction, geometric convection, point set simplification, Hybrid model, implicit surfaces, polygonal meshes, blending, local deformations.

Discipline Informatique.

Adresse du laboratoire

Laboratoire d'Informatique en Images et Systèmes d'Information (LIRIS) - UMR 5205
Université Claude Bernard – Lyon 1
Bâtiment Nautibus,
43 boulevard du 11 Novembre 1918
69622 VILLEURBANNE CEDEX
