



HAL
open science

Algorithmique et géométrie discrète pour la caractérisation des courbes et des surfaces

David Coeurjolly

► **To cite this version:**

David Coeurjolly. Algorithmique et géométrie discrète pour la caractérisation des courbes et des surfaces. Interface homme-machine [cs.HC]. Université Lumière - Lyon II, 2002. Français. NNT : . tel-00167370

HAL Id: tel-00167370

<https://theses.hal.science/tel-00167370v1>

Submitted on 20 Aug 2007

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

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



Distributed under a Creative Commons Attribution - NonCommercial - ShareAlike 4.0 International License

THÈSE

pour obtenir le grade de

DOCTEUR

en

INFORMATIQUE

présentée et soutenue publiquement par

David COEURJOLLY

le 18 DÉCEMBRE 2002

**Algorithmique et géométrie
discrète pour la caractérisation des
courbes et des surfaces**

préparée au sein du laboratoire ERIC

sous la direction de

Serge Miguet et Laure Tougne

COMPOSITION DU JURY

M.	Jean-Pierre Braquelaire	Rapporteur	(Professeur)
M.	Jean-Marc Chassery	Rapporteur	(Directeur de Recherche CNRS)
M.	Jean-Michel Jolion	Examineur	(Professeur)
M.	Jean-Pierre Reveillès	Examineur	(Professeur)
M.	Serge Miguet	Directeur de thèse	(Professeur)
Mme.	Laure Tougne	Directeur de thèse	(Maître de conférences)

Remerciements

Mes premiers remerciements iront à mes deux directeurs de thèse, tout d'abord Serge pour sa grande disponibilité, sa fonction *encyclo de l'imagerie* à laquelle j'ai eu recours de nombreuses fois, pour son appareil photo numérique aux fêtes d'équipe (certains pourraient remettre en question l'avantage d'instantanéité des appareils photos numériques...). Un grand merci à Laure à qui je décerne le prix de *Correcteuse de Fêtes* en chef (sans elle, ce manuscrit aurait une forme de manuscry) et qui a dû me supporter pendant trois ans en face d'elle (ça fait parfois mal aux oreilles...).



Je tiens à remercier Achille Braquelaire et Jean-Marc Chassery pour avoir accepté de rapporter sur ce loooooonng manuscrit. Je souhaiterais aussi remercier Jean-Michel Jolion d'avoir bien voulu participer à mon jury de soutenance.



Un grand merci à Jean-Pierre Reveillès, tout d'abord pour avoir fait parti de mon jury de soutenance, mais surtout pour son accueil très chaleureux. Je suis encore étonné du grand intérêt que Jean-Pierre a porté sur mes premiers travaux alors que j'étais thésard en première année, découvrant le monde de la recherche à un séminaire à Dagstuhl. Sa disponibilité n'a jamais défailli au cours de nos nombreuses rencontres en séminaire ou en conférence.

Je tiens à remercier toutes les personnes que j'ai rencontrées en conférences ou en séminaire. De nombreux développements sont nés de ces rencontres. Merci donc à Isabelle (les deux), Annick (à qui je dois une glace nougat-érable), *the swedish dream team* (Gunilla, Stina et Ingela). J'exprime aussi toute ma reconnaissance à Reinhard Klette pour la vision non franco-française de la géométrie discrète qu'il a su m'apporter. Grand merci à Jean-Bruno Brzoska, à Frédéric Flin et toute l'équipe du CEN de Météo-France pour m'avoir permis de confronter mes trucs avec la *réalité*, ainsi que pour les discussions toujours très intéressantes que l'on a pu avoir.

Ensuite, derrière toute thèse il y a un homme de l'ombre, technicien des basses œuvres (en gros c'est du débogage et des problèmes \LaTeX), le mien s'appelle David (aussi mais lui n'a pas encore vu la beauté des cubes... un jour peut-être...) et je le remercie bien bas. Petit coucou aussi aux membres du labo ERIC pour l'ambiance qui y règne, en vrac: les Tweed-Clech (Tiff a eu le prix d'aspirante *Correcteuse de Fêtes*), les Scuturici, les Jalam, Stephane et Fabrice (mes potes de galère)...

Pour terminer, merci à la famille pour leur support moral, en particulier à Jean-François (toujours avoir un frère stateux sous la main...), et bien sûr à Anne (toujours avoir une copine instit' sous la main...).



Résumé

Cette thèse se situe dans le cadre de la géométrie discrète qui constitue l'une des grandes familles de méthodes dédiées à l'analyse automatisée des formes dans les images numériques 2D et 3D. Tous les systèmes d'acquisition d'images fournissent des données organisées sur une grille régulière, appelées *données discrètes*. Les méthodes que nous nous proposons d'explorer et d'étendre conservent aux données ce caractère discret, par opposition aux techniques qui construisent préalablement un modèle continu approximant les objets à analyser.

Nous nous intéressons plus particulièrement à l'étude des courbes et des surfaces discrètes. Dans un premier temps, nous analysons les objets de base que sont les droites, les plans et les cercles discrets. Nous présentons des algorithmes qui permettent de les caractériser et proposons des extensions à ces méthodes. Ensuite, nous étudions des métriques sur les objets discrets comme la transformation en distance euclidienne ou la notion de géodésique discrète. Une approche basée sur la visibilité dans les domaines discrets est introduite. La troisième partie est consacrée à la définition et à l'évaluation d'estimateurs de mesures euclidiennes telles que la longueur, la courbure ou l'aire. Des résultats de convergence de ces estimateurs sont établis. Enfin, nous présentons les applications dans lesquelles ces recherches ont été utilisées : classification automatisée d'objets archéologiques et analyse des micro-structures d'échantillon de neige.

Mots clés : géométrie discrète, analyse d'images, algorithmique.

Abstract

The context of the work presented in this thesis is the digital geometry. This research area is devoted to the automatic analysis of objects in digital images in dimension 2 and 3. All acquisition devices provide data organized on regular grids, called *digital data*. The algorithms that are explored and extended keep the discrete aspect of the data, in opposition to techniques based on an approximation process of a continuous model.

More precisely, we are interested in the study of digital curves and surfaces. First of all, we consider basic digital objects such as digital straight lines, planes and circles. We present algorithms that allow to characterize such objects and we propose some extensions of these methods. Then, we study some metrics on the digital objects such as the Euclidean distance transform and the notion of digital geodesic. An approach based on the visibility property in digital domains is presented. In the third part, we define and evaluate estimators of the Euclidean measurements such as the length, the curvature or the area. Some results on the convergence of these estimators are presented. Finally, we illustrate some applications in which these researches have been used for: archaeological object automatic classification and snow sample micro-structure analysis.

Keywords: digital geometry, image analysis, algorithmic.

Table des matières

Introduction	1
I Géométrie discrète et objets discrets	5
1 Notions de base	7
1.1 Introduction	8
1.2 Espace discret et connexité	8
1.3 Objets, courbes, surfaces et hyper-surfaces	10
1.4 Codage et propriétés du codage des courbes discrètes	13
1.5 Processus de discrétisation	15
1.6 Conclusion	17
2 Droites et Plans discrets	19
2.1 Introduction	20
2.2 Droites discrètes 2D	20
2.3 Droites discrètes 3D	44
2.4 Plans Discrets	54
2.5 Statistique pour la reconnaissance de droites et plans discrets	73
2.6 Conclusion	78
3 Le cercle discret	79
3.1 Introduction	80
3.2 Synthèse et analyse du cercle discret	80
3.3 Reconnaissance et segmentation	83
3.4 Conclusion	101
II Analyse d'objets et mesures	103
4 Métriques discrètes	105
4.1 Introduction	107
4.2 Transformée en distance	107
4.3 Squelette de forme et axe médian	134
4.4 Visibilité et géodésiques discrètes	143

4.5	Conclusion sur ce chapitre	158
5	Mesures sur des objets discrets	161
5.1	Introduction	163
5.2	Contexte théorique : la convergence asymptotique	163
5.3	Tangentes et normales	165
5.4	Longueur et aire	178
5.5	Courbures d'objets discrets	199
5.6	Conclusion	220
	Conclusions et perspectives	223
III	Annexes	227
A	Applications	231
A.1	Introduction	232
A.2	Extraction de primitives pour la classification de profils de stèles funéraires	232
A.3	Analyse microscopique d'échantillon de neige	236
A.4	Modèle déformable discret	239
A.5	Conclusion	240
B	Preuves de convergence asymptotique	241
B.1	Introduction	242
B.2	Courbes en dimension d : approche par <i>tube</i>	242
B.3	Surfaces : aire et normales	247
C	Programmation linéaire : algorithme de PREPARATA et SHAMOS	253
D	Index des auteurs cités	257

Table des figures

1	Organisation du manuscrit, les flèches indiquent les dépendances entre chapitres.	3
1.1	Pavages réguliers en dimension 2 par carrés, par hexagones et par triangles	8
1.2	Représentation duale par maillage en dimension 2.	8
1.3	Différentes connexités en 3D sur un pavage par cubes.	9
1.4	Paradoxe de connexité pour un maillage par carré	11
1.5	Défaut de la notion de bord en tant qu'ensemble de points discrets	12
1.6	Décomposition d'un pixel et d'un voxel en cellules de dimensions inférieures.	13
1.7	Codage de FREEMAN	14
1.8	Codage de FREEMAN d'arcs	14
1.9	Codage de BRIBIESCA	15
1.10	Processus de discrétisation classiques	16
1.11	Discrétisation par supercourvertures	17
2.1	Décomposition de l'espace 2D en quadrants (<i>gauche</i>) et en octants (<i>droite</i>).	20
2.2	Seul l'arc (a) est un segment discret selon les définitions usuelles des droites discrètes (cf ci-dessous).	21
2.3	Propriétés de FREEMAN	22
2.4	Propriété de corde de ROSENFELD	22
2.5	Propriété de régularité de HUNG	23
2.6	Représentation graphique des quadruplets (n, q, p, s) de DORST et SMEULDERS pour la caractérisation des droites discrètes ; notre segment se code donc $(14, 5, 2, 2)$	24
2.7	Droite discrète arithmétique naïve de paramètres $\mathcal{D}(2, 5, -1, 5)$	24
2.8	Connexité des droites discrètes en fonction de ω	25
2.9	Droite discrète arithmétique naïve de paramètres $\mathcal{D}(2, 5, -1, 5)$: (<i>haut</i>) représentation par restes avec les points et droites d'appui (<i>bas</i>) construction par union de pointillés.	27
2.10	Pointillés et vecteur de Bezout pour la droite $\mathcal{D}(2, 5, -1, 5)$	28
2.11	Ensemble de pixels \mathcal{S} et domaine des droites euclidiennes $y = \alpha x + \beta$ associées, noté $\bar{\mathcal{S}}$	32
2.12	Passage dans l'espace des paramètres associés aux droites	32

2.13	Représentation de la droite discrète $\mathcal{D}(2, 5, -1, 5)$ sous forme d'intervalles et droites euclidiennes solutions.	33
2.14	Algorithme de O'ROURKE pour construire l'ensemble des droites réelles passant par une série d'intervalles.	33
2.15	Différentes formes du domaine des droites dans le dual.	35
2.16	Illustration des éventails de Farey introduits par MCILROY (1985)	36
2.17	L'ajout d'un nouveau point introduit une contrainte sur $\bar{\mathcal{S}}$ d'ordre q , le point B' a une abscisse qui est le médian des abscisses de A et B dans la serie de Farey d'ordre $q + 1$	37
2.18	Description des notations : U et U' sont les points d'appui supérieurs d'abscisses minimale et maximale, L et L' sont les points d'appui inférieurs d'abscisses minimale et maximale.	39
2.19	Différents cas de figure lrsde l'ajout d'un point à une droite discrète	40
2.20	Comparaisons entre approche arithmétique et approche basée sur l'espace des droites dans le dual	43
2.21	Exemples de segmentation d'une courbe discrète en droite discrète avec l'algorithme de DEBLED-RENNESON et REVEILLÈS (1995)	46
2.22	Droites discrètes 3D de paramètres $\mathcal{D}_{3D}(10, 7, 3, 0, -9, 13, 17)$, conformément au théorème 2.7, cette droite est 6-connexe. De plus, les droites 2D dans les plans Oxy et Oxz sont 4-connexes (DEBLED-RENNESON, 1995).	48
2.23	Droites discrètes 3D 26-connexes	48
2.24	Décomposition de l'espace en 48 ^{ème} , représentation graphique du groupe O_h (REVEILLÈS, 2001).	50
2.25	Exemples de segmentation de courbe 26-connexe en segments de droite naïve 3D	53
2.26	Illustration de la propriété du triangle corde avec un morceau de plan discret et trois triangles vérifiant cette propriété.	55
2.27	Illustration du théorème de KIM (1984b)	55
2.28	Cas négligé dans le théorème de KIM (1984b) présenté par DEBLED-RENNESON (1995)	56
2.29	Rectangle et rectangle partiellement étendu en x de VEELAERT (1994).	57
2.30	(a) plan arithmétique naïf $\mathcal{P}(6, 13, 17, 17)$ et (b) plan arithmétique standard $\mathcal{P}(6, 13, 17, 36)$ pour $0 \leq x, y \leq 10$	57
2.31	Morceau de plan discret $\mathcal{P}(7, 17, 57, 0)$ et représentation par reste pour $0 \leq x, y \leq 10$	58
2.32	Illustration des points d'appui d'un plan discret	59
2.33	Présence de droites discrètes dans des plans naïfs	60
2.34	Illustration des polygones d'appui pour un morceau de plan naïf $\mathcal{P}(7, 17, 57, 0)$	62
2.35	(a) Morceau de plan $P(1, 3, 4, 0)$ avec ses points d'appui et (b)-(c) différentes vues du polyèdre des plans euclidiens $\bar{\mathcal{E}}$ dont la discrétisation contient le morceau P	63
2.36	Illustration de la contrainte engendrée par un point d'appui v contenu dans le polygone d'appui de sommets notés $\{e^i\}_{i=1..4}$	66
2.37	Illustration du polyèdre avec la notion de double-cône	67

2.38	Enveloppe convexe du réseau engendré par la suite $\{(i, \{\frac{ai}{b}\})$ avec $0 < i < b\}$ pour $a = 5$ et $b = 17$	67
2.39	Illustration dans le cas 2D de la preuve du théorème 2.11	68
2.40	Condition nécessaire pour que le polyèdre des solutions dans l'espace des paramètres soit un double-cône	69
2.41	Facettisation basée sur l'enveloppe convexe 3D (a) et sur un algorithme de type Marching-Cubes (b).	70
2.42	Liens entre droite discrète et loi de BERNOULLI	74
2.43	Fréquences d'apparition des codes quand ceux-ci sont analysés en abscisse croissante.	74
2.44	Fréquences d'apparition des codes quand ceux-ci sont analysés suivant une abscisse croissante sur une courbe discrète formée de 4 segments de droite de 100 pixels et de pente $\frac{1}{4}, \frac{1}{2}, \frac{1}{4}$ et $\frac{2}{3}$	75
2.45	Test d'adéquation à une loi de BERNOULLI pour la reconnaissance de droite	76
2.46	Illustration du test statistique	76
2.47	Morceau de plan discret $\mathcal{P}(7, 17, 57, 0)$ et classification des différents surfels.	77
2.48	Quelques pistes pour la création d'un test d'adéquation à la loi empirique	78
3.1	Discretisation de cercles euclidiens	80
3.2	Cercle arithmétique de rayon 8.	81
3.3	Exemple de construction d'un cercle de Pitteway plancher de rayon 20.	82
3.4	Un exemple de courbe discrète (les points de rupture sont représentés en gris clair) et le code de rupture de ce segment est : 2-2-1-3-2.	85
3.5	Exemple de domaine des centres des cercles séparant deux ensembles de points du plan construit par intersection de contraintes issues des médiatrices.	88
3.6	Optimisation du calcul du cercle séparateur par réduction des ensembles S et T	89
3.7	Illustration des différentes définitions : un segment discret de vecteur directeur \vec{u} dans le premier octant, le vecteur de BEZOUT \vec{v} associé et les points de BEZOUT qui se trouvent sur la droite en pointillé et qui sont générés par \vec{u}	90
3.8	(a) séparation du segment $[pq]$ de tous les pixels au dessus grâce au point de BEZOUT (en gris) et (b) points de BEZOUT associés à une courbe polygonale convexe.	90
3.9	Illustration du choix du point de BEZOUT quand la pente de $[pq]$ n'est pas une fraction irréductible.	92
3.10	L'enveloppe convexe considérée dans le théorème 3.2 ne contient pas l'arête marquée d'une croix.	93
3.11	Exemple de segmentation en morceaux SCoC d'une courbe discrète	94
3.12	Exemple de calcul d'enveloppe convexe arithmétique inférieure d'un segment de droite discrète de paramètres (17, 28).	95
3.13	Distance euclidienne entre le centre du cercle et le centre de gravité de l' acd en fonction du nombre d'arêtes de Γ considérées pour un cercle de rayon 100.	99

3.14 Exemples d'évolution de l' <i>acd</i> sur un cercle de rayon 100 quand on considère un nombre croissant d'arêtes de Γ	99
3.15 Exemples de segmentation en arcs de cercles	100
3.16 Détails de la segmentation de la courbe en S	100
3.17 Comparaison entre notre approche et celle de KOVALEVSKY (1990) .	101
4.1 Boules de rayon 100 pour les métriques : (a) d_4 (ou distance de Manhattan), (b) d_8 (ou distance de l'échiquier) et (c) distance euclidienne.	108
4.2 Erreur de la transformée de DANIELSSON sur un voisinage en 4-connexité	109
4.3 Illustration de l'algorithme de SAITO et TORIWAKI en 2D	110
4.4 Diagramme de Voronoï et triangulation de Delaunay	112
4.5 Notations de BREU ET AL.	113
4.6 Illustration de l'algorithme de BREU ET AL.	114
4.7 Illustration de la proposition 4.1	115
4.8 Calcul du diagramme de Voronoï discret 2D avec suppression de sites	118
4.9 Construction du diagramme de Voronoï discret 3D	120
4.10 Diagramme de Voronoï et EDT de sites distribués aléatoirement . . .	125
4.11 Diagramme de Voronoï et EDT d'un disque	126
4.12 Diagramme de Voronoï et EDT de deux disques avec une intersection non vide	126
4.13 Diagramme de Voronoï et EDT d'un rectangle	127
4.14 Diagramme de Voronoï et EDT d'objets synthétiques	128
4.15 Illustration du diagramme de Voronoï discret 3D	129
4.16 Illustration du diagramme de Voronoï discret 3D et de la transformée en distance	129
4.17 Transformée en distance euclidienne 3D	130
4.18 Reconstruction par surface implicite à partir d'un squelette	130
4.19 Diagramme de Voronoï et EDT sur une grille anisotropique	132
4.20 Rééchantillonnage de la grille hexagonale pour le calcul du diagramme de Voronoï discret	133
4.21 Diagramme de Voronoï et EDT sur une grille hexagonale	133
4.22 Représentation surfacique de la transformée en distance : l'axe médian correspond <i>aux crêtes</i> de ces surfaces.	135
4.23 Squelette d'un polygone basé sur le diagramme de Voronoï	136
4.24 Illustration de la fonction d'étanchéité et l'angle bissecteur de ATTALI (1995).	138
4.25 Squelette d'un disque	139
4.26 Squelette de deux disques joints	140
4.27 Squelette d'un rectangle	141
4.28 Squelette d'une forme synthétique	142
4.29 Illustration sur le graphe des paramètres de la simplification du squelette par hysteresis.	143

4.30	Illustration des définitions : à gauche, tous les points dans la zone bleue sont visibles du point rouge. A droite, construction du graphe de visibilité et du plus court chemin (en gras) entre le point rouge et le point bleu.	144
4.31	Exemple de domaine $\mathcal{S}(s, t)$ des pixels $(0, 0)$ et $(3, 2)$	145
4.32	Domaine de visibilité associé à un ensemble de pixels	146
4.33	Illustration de la preuve du lemme 4.4	147
4.34	Structure de files de priorité : ensemble de files FIFO indicées par la distance au pixel source s	152
4.35	Analyse expérimentale de l'étiquetage en distance géodésique	155
4.36	Diagramme de visibilité discrète	156
4.37	Application de la distance géodésique en imagerie médicale	156
4.38	Illustration de la visibilité restreinte en 3D	157
4.39	Exemple d'étiquetage en distance géodésique 3D	157
4.40	Exemple d'étiquetage en distance géodésique 3D	158
4.41	Géodésiques surfaciques	159
5.1	(gauche) Polygones réguliers interne et externe d'un cercle pour $n_m = 6$, (droite) pourcentage d'erreur de l'estimation du périmètre avec le polygone régulier interne pour n croissant.	164
5.2	Objets multi-grilles proposés dans KLETTE ET AL. (1999) pour l'analyse d'estimateurs de longueurs.	165
5.3	Calcul de la tangente discrète par ajout symétrique de pixels et test d'appartenance à une droite discrète. Les pixels colorés terminent la tangente.	166
5.4	Illustration de l'algorithme de FESCHET ET TOUGNE (1999) pour le calcul en temps optimal des tangentes discrètes.	167
5.5	Analyse de la carte de distance pour l'estimation des normales	170
5.6	Comparaison entre les visualisations avec l'algorithme des Marching Cubes et avec les normales basées sur le GTD sur des formes géométriques simples.	171
5.7	Visualisation d'un échantillon de neige avec une estimation de normales basée sur l'algorithme des Marching-Cubes (a) et avec l'estimateur basé sur le GTD (b) (Images Météo-France CEN, ESRF, voir les explications dans le paragraphe A.3).	172
5.8	Comparaison de l'estimation de la normale entre l'algorithme basé sur le GTD et l'algorithme issu des Marching Cubes sur des sphères de taille croissante.	172
5.9	Illustration 2D de l'utilisation du diagramme de Voronoï discret pour l'estimation directe du GTD	173
5.10	Illustration des normales élémentaires associées à un morceau de surface discrète.	174
5.11	Approche de CHEN ET AL. (1985) pour l'estimation de la normale basée sur une décomposition cellulaire de la surface discrète.	174
5.12	Approche de PAPIER (1999) pour l'estimation de la normale	175

5.13	Exemple de calcul de normales basé sur une pondération de normales élémentaires avec un voisinage géodésique	176
5.14	Décomposition en coupes d'un volume selon les axes de la grille.	177
5.15	Décomposition en coupes d'une surface discrète composée de surfels	178
5.16	Illustration des algorithmes de segmentation en droites discrètes	180
5.17	Notations pour le calcul de polygone de longueur minimale (MLP)	181
5.18	Illustration des algorithmes SZ-MLP et AS-MLP	182
5.19	Approche AS-MLP : les sommets des bords de Jordan sont déplacés en fonction d'un paramètre δ	182
5.20	Illustration des vecteurs élémentaires à une courbe discrète.	183
5.21	Calcul de l'apport d'un pixel dans l'estimation de longueur par intersection entre le pixel et la tangente estimée en celui-ci.	184
5.22	Illustration des estimateurs de longueur basés sur l'intégration des normales discrètes	185
5.23	Convergence asymptotique expérimentale des estimateurs de longueur	188
5.24	Mesure d'efficacité de l'approximation polygonale pour l'estimation de longueur	189
5.25	Erreur relative sur un carré de taille fixe sur lequel une rotation est effectuée	189
5.26	Temps de calcul des différents estimateurs sur une station Sparc Ultra 10.	190
5.27	Illustration du calcul du polygone de longueur minimale associé à une courbe discrète 6-connexe	191
5.28	Analyse expérimentale de convergence asymptotique de l'estimateur \mathcal{E}_{3D-DSS}	192
5.29	Analyse expérimentale de convergence asymptotique de l'estimateur \mathcal{E}_{3D-DSS}	193
5.30	Volumes discrets multi-grilles considérés pour l'analyse expérimentale des estimateurs d'aire.	197
5.31	Convergence asymptotique expérimentale des différents estimateurs sur des sphères multi-grilles.	197
5.32	Convergence asymptotique expérimentale des différents estimateurs sur des cylindres multi-grilles.	198
5.33	Convergence asymptotique expérimentale des différents estimateurs sur des demi-sphères trouées multi-grilles.	198
5.34	Illustration de la fonction $a(v)$ représentant l'apport de chaque voxel v à l'aire globale de la surface : (a) sphère discrète, (b) sur un cube et (c) sur un cube lisse tourné.	199
5.35	Exemple de plans principaux (Π_1 et Π_2) à une surface avec leur cercle principal respectif et le repère de FRENET $(\vec{N}, \vec{b}, \vec{t})$ ainsi défini.	201
5.36	Caractérisation locale des surfaces en fonction des courbures moyennes et gaussiennes.	202
5.37	Exemple de calcul de courbure normalisée d'ordre m pour $m = 4$	203
5.38	Illustration de la première parabole des cercles arithmétiques et calcul de l'ensemble des cercles de hauteur l	205

5.39	Analogie en l'anneau euclidien $(r - 1/2, r + 1/2)$, la tangente au cercle de rayon r dans cet anneau et la tangente discrète en un point d'un cercle arithmétique.	205
5.40	Comparaison entre le graphe de courbure de VIALARD (1996a) (avec $\sigma = 1$) et l'estimation de l'algorithme 25	207
5.41	Comparaison entre le graphe de courbure de VIALARD (1996a) (avec $\sigma = 1$) et l'estimation de l'algorithme 25	207
5.42	Illustration de l'estimateur de courbure basé sur les demi-tangentes à gauche et à droite de p_i	209
5.43	Exemple de calcul de courbure sur une courbe discrète 3D avec une estimation du cercle osculateur	211
5.44	Illustration en 2D de l'estimateur de courbure moyenne de BULLARD ET AL. (1995).	213
5.45	Exemples de calcul de courbure moyenne avec l'estimateur de BULLARD ET AL. (1995)	214
5.46	Illustration de l'application de GAUSS dans le cas de surfaces dans \mathbb{R}^3	215
5.47	Application de GAUSS et estimation de courbure Gaussienne dans le cas d'une surface triangulée.	216
5.48	Illustration des différentes solutions pour l'estimation de l'aire d'un nuage de points sur une sphère	217
5.49	Exemples de calcul d'enveloppes convexes sphériques	218
5.50	Ellipse de meilleure approximation dans le plan. Les directions des axes principaux sont données par les vecteurs propres de la matrice d'inertie.	219
5.51	Exemples de calcul de courbure Gaussienne	221
5.52	Illustration du calcul du signe de la courbure Gaussienne	222
A.1	Exemples de profils de stèles funéraires sans pré-traitement.	233
A.2	Exemple de graphe des courbures	234
A.3	Vectorisation et reconstruction de profils de stèles funéraires	235
A.4	Codage d'une courbe discrète après segmentation par les points caractéristiques	235
A.5	Exemples de volumes binaires d'échantillons de neige	236
A.6	Exemples de coloriage des voxels à la surface d'échantillons de neige par une estimation de la courbure moyenne proposée par BRZOSKA ET AL. (1999b) (Météo-France/ESRF).	238
A.7	Exemples de déformation d'un contour discret.	239
B.1	Illustration du θ -agrandissement d'un tube T (a) et segmentation de γ en une séquence de tubes (b). Le paramètre θ doit être suffisamment petit pour éviter le cas (c).	243
B.2	θ -rétrécissement d'un tube T et différentes notations pour la preuve du lemme B.1. La figure (b) correspond à la projection 2D du pire cas pour le lemme.	244
B.3	Projection en 2D du pire cas et illustration des différentes notations pour la preuve du lemme B.2.	245
B.4	Illustration pour la preuve du théorème B.4 (projection 2D du pire cas).	248

B.5	Illustration de la construction de f_r dans le cas 2D	250
C.1	Différentes notations pour l'illustration de l'algorithme de PREPARATA et SHAMOS (1985).	254
C.2	Transformation dans l'espace dual et illustration de la proposition C.1.	254
C.3	L'enveloppe inférieure des droites issues de $\mathcal{E}^+ = \{e_i\}$ permettent de construire les sommets de la partie supérieure des points $\{e_i^*\}$. De même pour l'ensemble \mathcal{E}^-	255

Liste des tableaux

1.1	Liens entre notations de voisinage pour un pavage par hypercubes. . .	9
2.1	Transformations pour la généralisation du tracé de droite dans tous les octants.	30
2.2	Algorithmes et complexités pour la reconnaissance de droites discrètes. . .	45
3.1	Algorithmes et complexités pour la reconnaissance de cercles discrets. . .	84
3.2	Résultats numériques sur la reconnaissance de cercle sur un cercle discret de rayon 100.	99
5.1	Évaluation théorique des différents estimateurs et principales références bibliographiques.	186
5.2	Caractérisation locale des surfaces en fonction des courbures moyenne et gaussienne.	201
5.3	Estimateurs discrets de mesures euclidiennes.	225

Table des algorithmes

1	Tracé de droite de BRESENHAM (1965)	29
2	Tracé de droites arithmétiques naïves de REVEILLÈS (1991) et DEBLED-RENNESON (1995)	30
3	Algorithme de GRABINER modifié	38
4	Ajout d'une contrainte avec réduction arithmétique, algorithme de VITTONÉ modifié	38
5	Algorithme de reconnaissance de droite discrète de DEBLED-RENNESON (1995)	41
6	Tracé de droites discrètes 3D de COHEN et KAUFMAN (1991)	49
7	Tracé de droites discrètes 3D de DEBLED-RENNESON (1995)	49
8	Implémentation du groupe O_h pour le changement de 48 ^{ème} d'espace de REVEILLÈS (2001)	51
9	Reconnaissance de segment de droite discrète naïve 3D de DEBLED-RENNESON (1995)	52
10	Segmentation en segments de droites discrètes naïves 3D de DEBLED-RENNESON (1995)	52
11	Tracé de cercle de BRESENHAM (1977)	81
12	Tracé de cercle de PITTEWAY (1974)	82
13	Enveloppe convexe arithmétique d'un segment de droite discrète $[0, 0], [b, a]$	96
14	Reconnaissance globale de cercle discret (COEURJOLLY ET AL., 2001c)	97
15	Segmentation en arcs de cercle (COEURJOLLY ET AL., 2001c)	97
16	Prédicat CACHÉ_ PAR de BREU ET AL. (1995)	116
17	Procédure de suppression de sites de BREU ET AL. (1995)	116
18	Nouveau prédicat CACHÉ_ PAR_ 2D	118
19	Prédicat CACHÉ_ PAR_ 3D	119
20	Diagramme de Voronoï discret en 3D	120
21	Diagramme de Voronoï discret en nD	124
22	Classe d'équivalence de visibilité à partir d'une source s	149
23	Classe d'équivalence de visibilité restreinte à partir d'une source s	150
24	Étiquetage en distance géodésique à partir d'une source s	154
25	Calcul de courbure par estimation de cercles osculateurs discrets	206
26	Calcul de courbure par reconnaissance de cercles osculateurs discrets	208
27	Calcul de courbure par demi-tangentes discrètes	210
28	Calcul de courbure par demi-tangentes discrètes	212
29	Aire d'un nuage de points sur la sphère par enveloppe convexe sphérique	218

TABLE DES ALGORITHMES

30	Aire d'un nuage de points sur la sphère par calcul d'une ellipse de meilleure approximation	220
31	Calcul de l'intersection de contraintes linéaires par l'algorithme de PREPARATA et SHAMOS (1985)	256

Introduction

Tous les systèmes d'acquisition de données images fournissent des données organisées sur une grille régulière, appelées *données discrètes*. Que ce soit pour une visualisation ou pour l'extraction de mesures sur ces objets discrets (paramètres de formes), les axiomes et théorèmes de la géométrie euclidienne ne sont pas directement applicables. Deux solutions s'offrent à nous. La première consiste à plonger les données discrètes dans un espace continu où ces théorèmes et mesures sont définis (par exemple en utilisant des processus d'interpolation). Une seconde alternative se base sur une transposition de ces théorèmes et mesures dans l'espace discret. Ces différentes re-définitions donnent lieu à un paradigme mathématique et informatique appelé *géométrie discrète*.

Un premier intérêt d'une telle définition est que nous évitons ainsi un changement de modèle coûteux en temps et potentiellement source d'imprécisions. Dans le cadre de la géométrie algorithmique, les erreurs d'arrondi engendrées par de tels processus peuvent affecter non seulement les résultats mais aussi la terminaison même de l'algorithme. Dans l'introduction de la thèse d'état de JEAN-PIERRE REVEILLÈS (1991), nous pouvons lire :

De façon générale, dès que des données ont été tronquées plus aucun théorème géométrique n'est vrai. On peut dire alors que la géométrie, qui, sur le papier, était déjà l'art de raisonner juste sur des figures fausses, devient, sur machine, l'art de raisonner juste à partir de calculs faux produisant des figures fausses, si toutefois les calculs terminent.

D'un point de vue algorithmique, l'analyse de ces objets sur une grille régulière à l'aide d'une méthodologie en nombres entiers permet, non seulement une gestion de ces incertitudes, mais aussi l'écriture d'algorithmes très efficaces. En effet, comme nous le verrons par la suite, la manipulation de nombres entiers et les liens très forts entre les objets discrets et des théorèmes de la théorie des nombres ou d'arithmétique offrent des possibilités d'implémentations d'outils géométriques très efficaces.

Derrière ces considérations d'ordre théorique se cache un intérêt pratique très important. En effet, une théorie géométrique et une algorithmique adaptée aux images nous permettent d'accélérer les processus mais surtout de proposer des outils d'analyse de formes cohérents avec la modalité des objets analysés.

Cet intérêt pratique a été illustré depuis très longtemps. Ainsi, en 1771, en présentant un principe d'interpolation linéaire basé sur des entiers et sur des modulus (ce qui correspond exactement aux fondements des droites discrètes), JEAN BERNOULLI conclut son article avec la certitude d'avoir présenté une technique qui allait être un outil de base de la géométrie discrète contemporaine :

[...] je me flatte de mettre sous les yeux des astronomes différentes méthodes d'abrèger considérablement leurs calculs, qu'ils ne connaissaient

pas encore, ou que, par des raisons qui cesseront, ils ont négligé de saisir lorsqu'elles leur ont été présentées pour la première fois ; [...]

Les recherches présentées dans ce manuscrit s'inscrivent dans ce cadre de la géométrie discrète et concernent plus particulièrement l'étude des courbes et des surfaces discrètes.

Organisation du manuscrit

Ce manuscrit est décomposé en trois parties : la première concerne les notions de base et les différents objets du modèle discret. La seconde s'intéresse à l'analyse de formes et aux mesures sur ces objets discrets. Enfin, la troisième partie correspond aux annexes.

Plus précisément, après avoir présenté quelques notions de base en géométrie utiles pour la suite, nous analysons les objets de base que sont les droites, les plans et les cercles discrets, et présentons les algorithmes qui permettent de les caractériser. En particulier, nous proposons une synthèse des différentes approches et algorithmes associés aux droites discrètes en dimension 2 et 3. Nous reprenons ce travail bibliographique sur les plans discrets en proposant une première caractérisation de la pré-image d'un morceau de plan discret. Nous concluons ce chapitre sur les droites et les plans par une analyse statistique de ces objets.

Nous nous intéressons par la suite aux notions de cercles discrets et aux différents algorithmes de reconnaissance associés. Nous proposons dans ce chapitre un algorithme de reconnaissance de cercle discret et de segmentation d'une courbe discrète en arcs de cercle basé sur une résolution arithmétique du problème.

Dans la seconde partie, nous utilisons cette analyse des objets de base pour caractériser des formes ou des contours de formes. Ainsi, nous présentons un algorithme de calcul de transformée en distance euclidienne sans erreur basé sur la notion de diagramme de Voronoï discret. En utilisant ce diagramme particulier, nous présentons des premières pistes vers une extraction de squelette de forme. Dans ce même chapitre, nous proposons une définition de la visibilité discrète dans des domaines discrets non-convexes basée sur la notion de droite discrète. Cette définition de la visibilité nous permet par la suite de proposer un calcul de chemins géodésiques discrets.

Ensuite, nous nous intéressons aux re-définitions, dans le modèle discret, de mesures euclidiennes. En utilisant le cadre théorique de la convergence asymptotique, nous présentons différents algorithmes permettant d'estimer la longueur d'une courbe discrète, les normales en tout point de cette courbe et la courbure associée à chaque pixel. Dans le cas des surfaces discrètes, nous proposons différents estimateurs permettant d'estimer l'aire de la surface, les vecteurs normaux en tout point et une estimation de différentes courbures (moyenne et gaussienne).

Finalement, la dernière partie correspond aux annexes et contient un premier chapitre présentant les différentes applications qui ont inspiré ces développements théoriques, un second contenant les différentes preuves de convergence asymptotique ainsi qu'un troisième détaillant un algorithme de programmation linéaire que nous utilisons dans les parties précédentes.

La figure 1 présente les différentes dépendances entre ces chapitres et parties. Par exemple, les notions introduites dans le chapitre sur les droites et plans discrets seront utilisées pour construire des estimateurs de mesures euclidiennes et nous permettent de définir la notion de visibilité discrète.

Une lecture linéaire des chapitres de ce manuscrit est possible, cependant nous pouvons conseiller au lecteur de commencer par le chapitre relatant les différentes applications (annexe A) afin de débiter par ce qui a motivé nos recherches.

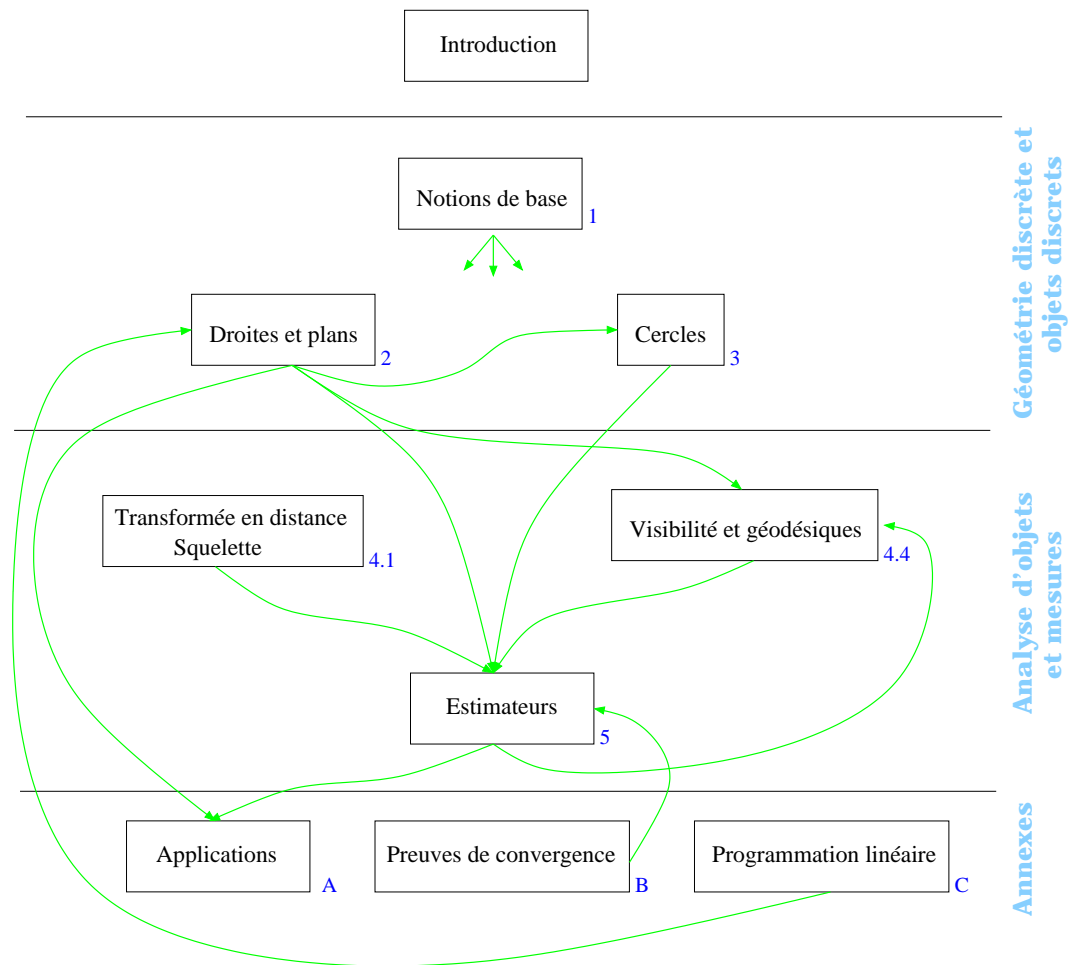
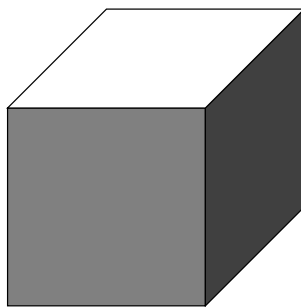


FIG. 1 — Organisation du manuscrit, les flèches indiquent les dépendances entre chapitres.

Première partie

Géométrie discrète et objets
discrets



Ceci est :

*un morceau de droites discrètes
un morceau de plan discret
un morceau de cercle discret*

parfois un simple cube mais c'est plus rare...

Chapitre 1

Notions de base

Sommaire

1.1	Introduction	8
1.2	Espace discret et connexité	8
1.3	Objets, courbes, surfaces et hyper-surfaces	10
1.3.1	Notions de topologie discrète	10
1.3.1.1	Approche basée sur les points discrets	12
1.3.1.2	Approche basée sur les éléments de surface	12
1.4	Codage et propriétés du codage des courbes discrètes	13
1.5	Processus de discrétisation	15
1.5.1	Discrétisations classiques	16
1.5.2	Discrétisation de simplexes	16
1.6	Conclusion	17

1.1 Introduction

Dans ce chapitre, nous introduisons certaines notions et propriétés de bases de la géométrie discrète. Celles-ci seront utilisées par la suite pour définir différents objets et algorithmes sur la grille discrète.

1.2 Espace discret et connexité

Nous commençons par décrire les notions de base en géométrie discrète (voir CHASERY et MONTANVERT 1991). Nous appelons *espace discret* par la suite un pavage régulier du plan en dimension 2 ou plus généralement de l'espace en dimension supérieure (voir figure 1.1). Nous appelons aussi un *point discret* le centre de gravité de chaque cellule dans le pavage considéré.

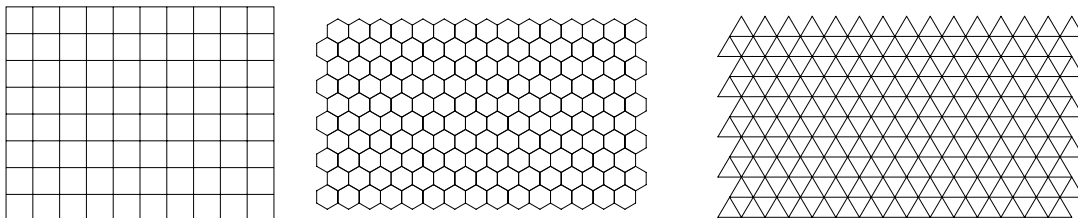


FIG. 1.1 –: Pavages réguliers en dimension 2 par carrés, par hexagones et par triangles

Dans ce qui suit, nous représentons graphiquement l'espace discret, soit par le pavage régulier, soit par le maillage issu de ce pavage qui est un graphe dont les sommets sont les points discrets et dont les arêtes représentent les adjacences des éléments du pavage (voir figure 1.2).

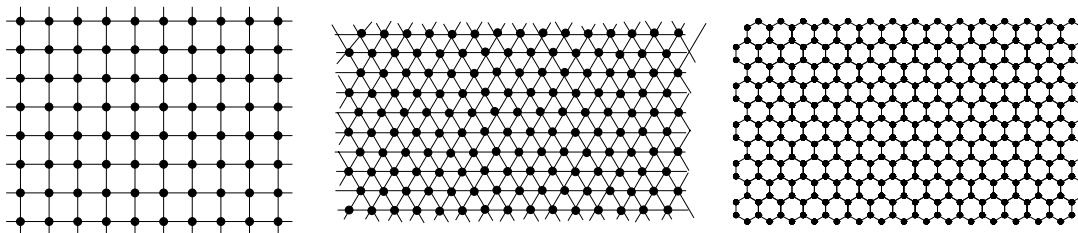


FIG. 1.2 –: Représentation duale par maillage en dimension 2.

En géométrie discrète, les espaces discrets engendrés par des carrés ou des cubes multi-dimensionnels pour les dimensions supérieures, sont les plus utilisés. Les points discrets associés à ces espaces sont donc des points de \mathbb{Z}^n pour la dimension n .

Sur cet espace discret, nous introduisons des notions de *voisinage* nous permettant

de construire des graphes d'adjacence. Ainsi, si nous considérons un espace discret 2D généré par des carrés, nous définissons le *4-voisinage* comme étant la relation d'adjacence par arêtes dans la partition de l'espace et le *8-voisinage* comme étant la relation d'adjacence par arêtes et par sommets. Plus formellement, deux points p et q de \mathbb{Z}^2 sont *4-voisins* (ou *4-adjacents*) si :

$$|x_p - x_q| + |y_p - y_q| = 1$$

De la même manière, deux points p et q de \mathbb{Z}^2 sont *8-voisins* (ou *8-adjacents*) si :

$$\max(|x_p - x_q|, |y_p - y_q|) = 1$$

En dimension 3, nous pouvons introduire les notions de *6-*, *18-* ou *26-voisinage* en considérant les adjacences par faces, par arêtes et par sommets (voir figure 1.3). Ces adjacences peuvent s'écrire formellement :

adjacence	caractérisation
6	$ x_p - x_q + y_p - y_q + z_p - z_q = 1$
18	p et q sont 26-voisins et $ x_p - x_q + y_p - y_q + z_p - z_q \leq 2$
26	$\max(x_p - x_q , y_p - y_q , z_p - z_q) = 1$

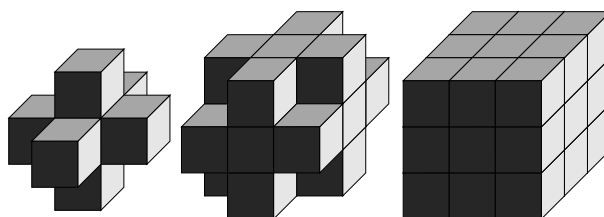


FIG. 1.3 – Différentes connexités en 3D sur un pavage par cubes.

En dimension n , une écriture unifiée de ces différentes relations d'adjacence s'exprime sous la forme de *O(r)-voisinage* : deux points de \mathbb{Z}^n sont *O(r)-voisins* si leurs coordonnées ne diffèrent que de 1 et qu'au moins $n - r$ coordonnées sont égales (voir JONAS et KIRYATI (1997) par exemple). Ainsi, les connexités 6, 18 et 26 en 3D s'écrivent respectivement $O(1)$, $O(2)$ et $O(3)$ dans ce schéma (voir table 1.1 pour une comparaison des différentes notations).

$O(r)$ -voisinage	k -voisinage		
	2D	3D	nD
1	4	6	$2n$
2	8	18	$2n^2$
3	.	26	...
...
r	.	.	$\sum_{i=n-r}^{n-1} \frac{n!}{i!(n-i)!} 2^{n-i}$
...
n	.	.	$3^n - 1$

TAB. 1.1 – Liens entre notations de voisinage pour un pavage par hypercubes.

Dans le cas d'un pavage hexagonal, nous n'avons qu'un seul voisinage élémentaire appelé *6-voisinage*. Pour une grille triangulaire, nous pouvons décrire, avec les mêmes principes que précédemment, le *3-voisinage* (connexité par arête) et le *12-voisinage* (connexité par arêtes et par sommets).

1.3 Objets, courbes, surfaces et hyper-surfaces

Maintenant que l'espace de travail est défini, nous pouvons décrire les objets que l'on manipulera par la suite. Bien évidemment, la notion de courbe ou de surface est intimement liée aux notions topologiques que nous introduirons sur l'espace discret.

Dans un premier temps, les relations d'adjacence décrites précédemment permettent de définir un k -chemin sur l'espace discret :

Définition 1.1 (k -chemin) *Soit un ensemble \mathcal{E} de points discrets et une relation de k -adjacence, \mathcal{E} est un k -chemin si pour tous les éléments $\{p_i\}_{i=0..n}$ de \mathcal{E} , p_i est k -voisin de p_{i-1} .*

Nous pouvons définir de la même manière un objet dans un espace discret muni d'une relation d'adjacence.

Définition 1.2 (k -objet) *Soit un ensemble \mathcal{E} de points discrets et une relation de k -adjacence, \mathcal{E} est un k -objet si pour tout couple p et q de \mathcal{E} , il existe un k -chemin dans \mathcal{E} .*

Selon une approche orientée graphe, un k -objet peut-être vu comme une composante connexe dans le graphe dont les sommets sont les points discrets et les arêtes les k -adjacences (ROSENFELD, 1970).

La notion de k -chemin décrite ci-dessus est très générale. Nous restreignons cette notion pour pouvoir introduire des propriétés topologiques sur ces objets. Nous proposons ainsi la notion de k -arc définie par ROSENFELD (1974).

Définition 1.3 (k -arc) *Soit un ensemble \mathcal{E} de points discrets et une relation de k -adjacence, \mathcal{E} est un k -arc si pour tous les éléments $\{p_i\}_{i=0..n}$ de \mathcal{E} , p_i a exactement deux points k -voisins, sauf p_0 et p_n appelés alors extrémités de l'arc.*

Si le k -arc est fermé on parle alors de k -courbe.

Définition 1.4 (k -courbe) *Soit un ensemble \mathcal{E} de points discrets et une relation de k -adjacence, \mathcal{E} est une k -courbe si \mathcal{E} est un k -arc et que $p_0 = p_n$.*

Pour simplifier l'écriture, nous utiliserons le terme générique *courbe discrète* pour représenter une k -courbe ou un k -arc.

L'introduction des courbes lève des problèmes topologiques. En effet, une propriété *souhaitable* sur une k -courbe serait que l'on puisse définir un intérieur et un extérieur. Ces définitions sont liées à la notion de propriété de Jordan en géométrie différentielle classique que nous détaillerons dans la section suivante.

1.3.1 Notions de topologie discrète

En géométrie différentielle classique, le théorème de séparabilité de Jordan-Brouwer (JORDAN, 1887) s'énonce :

Théorème 1.1 (Théorème de Jordan-Brouwer) *Pour toute variété M , C^∞ et de dimension $n - 1$ dans \mathbb{R}^n , nous avons :*

- M est orientable, c'est-à-dire qu'il existe un champ de vecteurs dont la différentielle ne s'annule pas, est continue et définie sur M , à valeurs dans \mathbb{R}^n ;

- M partage l'espace en deux domaines (ouverts connexes) dont l'un est borné et l'autre non et dont M est la frontière commune ;
- il n'existe pas de chemin continu entre ces domaines qui n'intersecte pas M .

En d'autres termes, une courbe fermée infiniment dérivable dans \mathbb{R}^2 définit un intérieur et un extérieur, et l'union de ces deux ensembles et de la courbe est une partition de \mathbb{R}^2 .

Afin de proposer une version discrète de ce théorème, une idée possible est de regarder la transposition directe de celui-ci en considérant des k -courbes et des k -arcs dans l'espace discret. Cependant, la figure 1.4 illustre le paradoxe de connexité dans le cas d'un pavage par carrés du plan. Sur la partie gauche de la figure, les pixels noirs forment une 8-courbe mais il existe un 8-chemin entre l'extérieur et le pixel blanc intérieur à la courbe. La propriété de séparabilité n'est donc pas vérifiée dans ce cas. Sur la partie droite de la figure 1.4, nous observons le cas où une 4-connexité est utilisée pour les pixels blancs et les pixels noirs, dans ce cas il n'existe pas de 4-chemin joignant les pixels blancs au pixel central mais la courbe des pixels noirs est elle déconnectée.

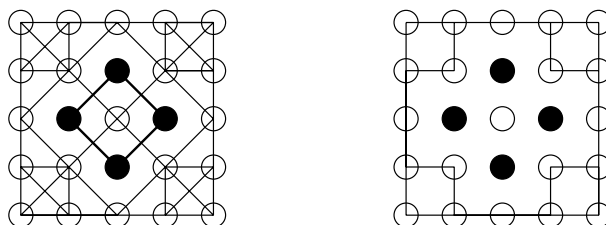


FIG. 1.4 –: Paradoxe de connexité pour un maillage par carrés, les arêtes de ces graphes représentent les différentes connexités choisies (voir texte).

Une façon simple pour résoudre le paradoxe en 2D pour une grille carrée est de considérer une certaine connexité pour la courbe et une autre pour son complémentaire. Ainsi, nous pouvons définir un analogue discret du théorème de Jordan.

Théorème 1.2 (Séparabilité de ROSENFELD) Dans \mathbb{Z}^2 , nous avons :

- le complémentaire de tout 4-arc (resp. un 8-arc) est 8-connexe (resp. 4-connexe) ;
- le complémentaire de toute 4-courbe (resp. une 8-courbe) a deux composantes 8-connexes (resp. 4-connexes) dont l'une est bornée et l'autre non.

La garantie d'un certain nombre de propriétés topologiques sur les objets discrets est indispensable pour deux raisons principales, d'une part les objets manipulés sont plus formalisés et mieux contrôlés, mais aussi cela permet l'écriture d'algorithmes très efficaces pour, par exemple, le parcours des éléments d'une courbe discrète ou d'une surface discrète. Il est donc nécessaire de choisir un cadre topologique nous permettant de résoudre ces problèmes.

Pour cela, deux grandes approches sont possibles, la première appelée *topologie digitale* (*digital topology* en anglais) considère des bords ou des surfaces discrètes comme étant des ensembles de points discrets (pixel ou voxels en 2D et 3D pour des pavage par hypercubes). La seconde se base sur une décomposition en éléments de dimension inférieure appelée *complexe cellulaire*.

1.3.1.1 Approche basée sur les points discrets

Dans cette approche, la surface ou le contour d'un objet discret est un ensemble connexe de point discrets. On parle alors de *bord* ou de *frontière* (*boundary* en anglais). Une première définition constructive proposée par MORGENTHALER et ROSENFELD (1981) permet de caractériser localement une telle surface dans le cas d'un espace discret dans \mathbb{Z}^3 . L'idée principale est de vérifier qu'il y a bien séparation locale dans un 26-voisinage entre le bord et le complémentaire. Cette caractérisation locale permet cependant d'énoncer un théorème global de Jordan pour ce type de surface.

Une définition de surface multidimensionnelle et un algorithme de parcours ont été proposé par UDUPA ET AL. (1982) et reposent sur une idée simple : un point discret appartient au bord de l'objet s'il possède un voisin dans le complémentaire (au sens de l'adjacence attribuée à celui-ci).

Une autre définition de ce type de surface basée sur des caractérisations locales a été proposé par BERTRAND et MALGOUYRES (1996); MALGOUYRES (1997). Dans ce cadre-là, une version discrète du théorème de Jordan peut être prouvée.

Un problème lié à ce type de surface a été illustré par PERROTON (1994) et est présenté dans la figure 1.5. Tout d'abord, le bord de l'objet et le bord de son complémentaire ne coïncident pas. De plus, une autre propriété classique de topologie n'est pas vérifiée : dans l'exemple de la figure 1.5, le pixel du bord indiqué par une flèche est voisin des deux composantes connexes de son complémentaire, et les deux surfaces que l'on souhaiterait (l'une bornant la cavité et l'autre le complémentaire non-borné) se trouvent fusionnées.

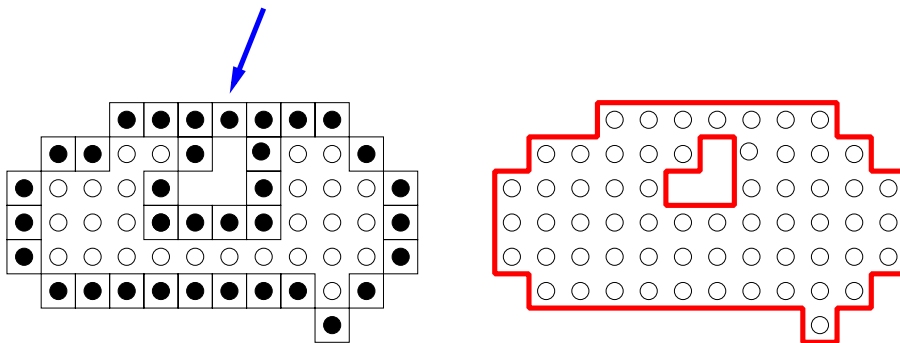


FIG. 1.5 – Défaut de la notion de bord en tant qu'ensemble de points discrets (points noirs à gauche) et surface de lignes basée sur une approche cellulaire (polygones en gras à droite).

1.3.1.2 Approche basée sur les éléments de surface

Dans cette approche, des éléments (appelés *cellules*) de dimension inférieure au polygone pavant l'espace discret sont considérés (voir figure 1.6). Ainsi, les cellules de dimension 0 sont les *pointels*, les éléments de dimension 1 sont appelés *lignels*, les faces ou éléments de dimension 2 sont appelés *surfels*. Cette décomposition ainsi que la théorie topologique associée correspond à un cas particulier de complexe simplicial baptisé *complexe cellulaire* pour des grilles dans \mathbb{Z}^n (KOVALEVSKY, 1990).

En se basant sur ce schéma de décomposition, ARTZY ET AL. (1981) et UDUPA (1994) ont proposé une définition de surface discrète dans \mathbb{Z}^n comme étant un en-

semble connexe (au sens d'une certaine relation d'adjacence) de cellules de dimension $n - 1$ "partageant" un point discret de l'objet et un point discret de son complémentaire. Par exemple en dimension 3, la surface discrète est un ensemble de *surfels* et ne sont considérés que les surfels partageant un voxel de l'objet et un voxel du fond. UDUPA *et al.* considèrent donc une $\kappa\lambda$ -surface discrète définie par la paire de connexités dite de Jordan (κ, λ) . Cette paire de connexités est une extension 3D de celle utilisée dans le théorème de séparabilité de ROSENFELD. Ainsi, une surface définie par une paire (κ, λ) considère un objet κ -connexe et un fond λ -connexe. Dans le cas 2D, les paires $(4, 8)$ et $(8, 4)$ sont dites de Jordan. En 3D nous avons les paires $(6, 18)$, $(18, 6)$, $(6, 26)$ et $(26, 6)$.

La relation d'adjacence entre les éléments de surfaces¹ composant la $\kappa\lambda$ -surface permet de construire des algorithmes d'extraction très efficaces de surfaces discrètes en dimension quelconque (voir PERROTON 1994 par exemple pour le cas 3D).

Dans une approche purement complexe cellulaire, KOVALEVSKY (1999) a proposé une définition et un algorithme de parcours de surfaces discrètes.

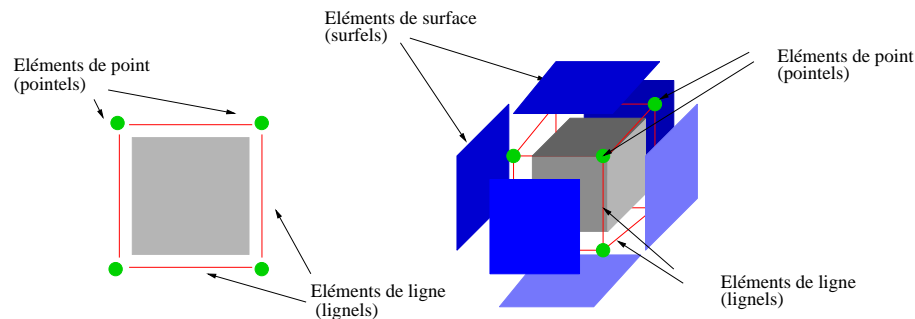


FIG. 1.6 —: Décomposition d'un pixel et d'un voxel en cellules de dimensions inférieures.

1.4 Codage et propriétés du codage des courbes discrètes

Dans cette partie, nous introduisons les codages usuels pour la représentation de courbes dans un espace discret. L'objectif est ici de proposer un codage de la courbe discrète ayant les propriétés de :

compacité : compression du codage par rapport à une liste de coordonnées des points discrets

réversibilité : possibilité de reconstruire la courbe à partir du codage

invariance : dans quelle mesure le code est-il invariant par certaines transformations géométriques ?

informative : Est-il possible d'extraire des informations au sens large sur la courbe à partir du codage (par exemple sa longueur...)?

Un codage usuel d'une courbe discrète est celui de FREEMAN (1961) qui consiste à introduire k labels pour les connexités possibles d'une k -courbe (voir 1.7 pour les codes 2D). Ensuite le codage consiste à associer le code qui correspond à chaque

1. relation β dans UDUPA (1994)

déplacement en k -connexité dans la courbe discrète. Il suffit donc de mémoriser les coordonnées du point de départ et le code de chaque déplacement pour reconstruire la courbe.

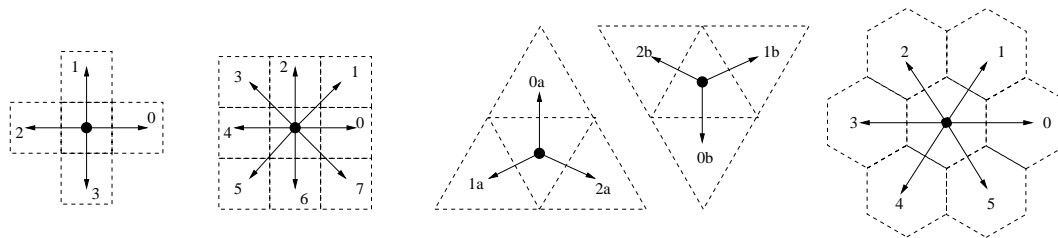


FIG. 1.7 – Codage de FREEMAN pour la 4- et la 8-connexité du pavage par carré, les codages pour les 3-connexité pour le pavage triangulaire et la 6-connexité pour la grille hexagonale.

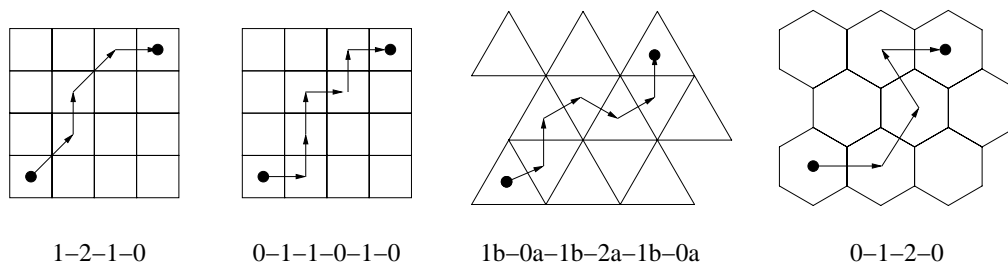


FIG. 1.8 – Codage de FREEMAN d’arcs respectivement 8-connexe, 4-connexe, 3-connexe et 6-connexe en utilisant les codes de la figure 1.7.

Ce codage de FREEMAN a de nombreuses propriétés (CHASSERY et MONTANVERT, 1991) : il permet notamment de segmenter une courbe discrète en quadrants, octants, sextants ... en fonction de l’espace discret. On peut aussi voir des transformations simples telles que la translation, la symétrie d’axes, la rotation, comme des processus de réécriture sur le code. Par exemple, pour une rotation de $\pi/2$ dans le sens trigonométrique d’une courbe 8-connexe, il suffit de remplacer le code $(\{c_i\}, (x, y))$ tel que les c_i sont les éléments du code et (x, y) sont les coordonnées du point de départ, par :

$$\begin{cases} c'_i = c_i + 2 \pmod{8} \\ (x', y') = (y, x) \end{cases}$$

Dans un objectif de comparaison de courbes discrètes par leur code, le codage de FREEMAN souffre du problème de variance par symétrie d’axe.

D’autres codes existent, ceux-ci diffèrent de l’application que l’on en fait. Nous présentons, par exemple, un codage très simple qui est lui invariant par symétrie. BRIBIESCA (1999) propose un codage de chaque *pointel* de la surface de l’objet par le nombre de points discrets de l’objet auxquels ce pointel est adjacent (voir figure 1.9). Ce code a les propriétés suivantes :

- invariance au choix du point de départ ;
- invariance par rotations d’angles $k\pi/2$ (avec $k \in \mathbb{Z}$) ;

- invariance par symétrie d'axe ;
- permet le calcul de la longueur au sens de FREEMAN ;
- permet le calcul du code du complémentaire par simple réécriture ;
- transposition possible entre ce code et le codage de ; FREEMAN, la propriété de synthèse est donc vérifiée ;
- extraction des caractéristiques d'EULER à partir du code.

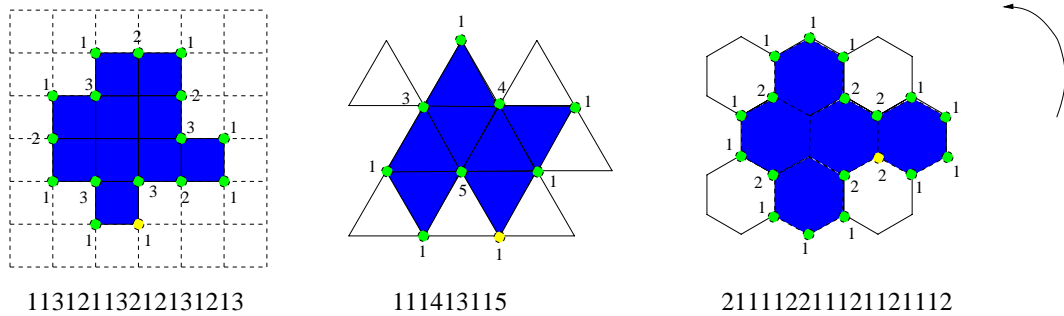


FIG. 1.9 — : Codage de BRIBIESCA pour la 4- et la 8-connexité du pavage par carré, les codages pour les 3-connexité pour le pavage triangulaire et la 6-connexité pour la grille hexagonale.

1.5 Processus de discrétisation

Pour terminer cette introduction des notions de base en géométrie discrète utiles pour la suite, nous détaillons les différents processus de discrétisation que nous serons amenés à utiliser.

Le contexte est le suivant : nous considérons un objet continu \mathcal{O} ou uniquement son bord, noté γ , et nous souhaitons construire l'ensemble des points discrets qui sont les *discrétisés* soit de \mathcal{O} , soit de γ . Dans ce qui suit, nous nous intéressons plus particulièrement au pavage par carrés et donc à des grilles dans \mathbb{Z}^n .

JONAS et KIRYATI (1997) proposent un ensemble de propriétés permettant une comparaison théorique de ces processus de discrétisation :

Permutation du système d'axes : est-ce que la discrétisation du symétrique par rapport à un axe de γ est le symétrique de la discrétisation de γ ?

Invariant par translation entière : le processus est-il dépend d'une origine sur la grille ?

Perturbation locale : est-ce qu'une petite perturbation locale de γ n'entraîne qu'une perturbation locale sur la discrétisation ?

Droite-arc : est-ce que la discrétisation d'une droite est un k -arc (pour une relation d'adjacence donnée) ?

Propriété de projection : pour les dimensions supérieures, est-ce le discrétisé du projeté sur un hyper-plan canonique de la grille coïncide avec le projeté du discrétisé ?

Convergence en distance : est-ce que la discrétisation converge selon une certaine métrique vers la courbe ou l'objet continu pour une résolution de grille croissante ?

1.5.1 Discrétisations classiques

Dans un premier temps, nous considérons que la courbe γ en position générale. C'est-à-dire qu'elle a une probabilité nulle de passer par un sommet du pavage de l'espace discret et de croiser une arête du maillage à égale distance des deux sommets adjacents à celle-ci. Nous verrons par la suite des processus de discrétisation qui gèrent ces cas pathologiques.

Un premier processus de discrétisation a pour analogie le tracé de droite de BRESENHAM (1965) : lorsque l'on a le choix entre deux points discrets, c'est-à-dire à chaque fois que la courbe γ croise une arête du maillage discret, on choisit celui qui est le plus proche. Ce processus s'appelle *Grid Intersect Quantization* (ou GIQ) et est illustré sur le premier schéma de la figure 1.10.

S'il on considère maintenant la discrétisation d'un objet et non d'une courbe, nous pouvons définir les discrétisations *interne* et *externe* de l'objet. Dans ce cas, lors du choix entre deux points discrets, nous choisissons respectivement celui qui est dans l'objet ou celui qui est extérieur à celui-ci (voir figure 1.10). Ces discrétisations sont appelées respectivement *Object Boundary Quantization* (OBQ) et *Background Boundary Quantization* (BBQ).

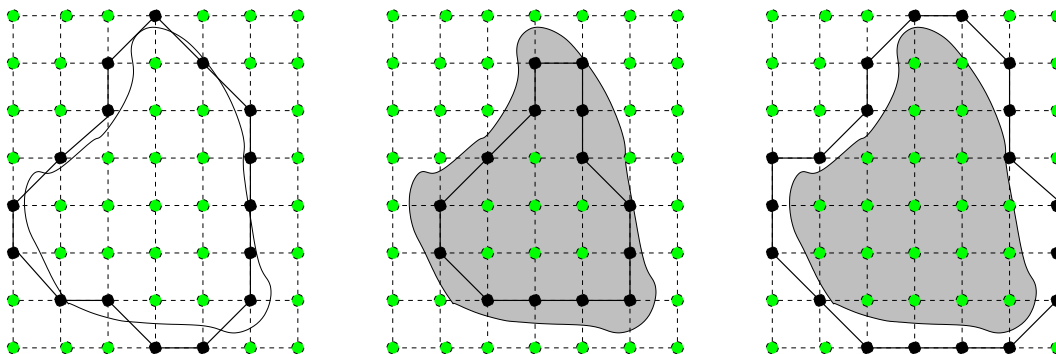


FIG. 1.10 – : *Processus de discrétisation classiques : (de gauche à droite) Grid Intersect Quantization, Object Boundary Quantization et Background Boundary Quantization.*

Ces discrétisations simples peuvent être facilement étendues pour des grilles dans \mathbb{Z}^n (voir JONAS et KIRYATI 1997 par exemple pour la 3D).

1.5.2 Discrétisation de simplexes

Dans ce paragraphe, on s'intéresse au cas particulier de la discrétisation de points, de droites, de plans ou d'hyper-plans en dimension n ; ou encore de m -simplexes (facettes de dimension m dans un espace de dimension n). Pour ce type d'objet, les cas pathologiques ne peuvent pas être supprimés et doivent être pris en compte.

De plus, certains de ces modèles permettent d'avoir des écritures analytiques des discrétisations des objets. C'est le cas notamment de la notion de *supercouverture* (voir CHASSERY et MONTANVERT 1991 par exemple) dont la description analytique pour des m -simplexes en dimension n a été proposée par ANDRÈS (voir ANDRÈS

2000 par exemple). Dans ce modèle, les cas pathologiques décrits précédemment s'appellent des *bulles* (voir figure 1.11).

ANDRÈS (2002) a aussi récemment proposé un autre modèle, appelé *standard*, permettant aussi une écriture analytique des objets discrétisés et ne contenant plus de *bulles* par le biais de convention d'orientations dans \mathbb{Z}^n .

Dans une approche plus topologique, COUPRIE ET AL. (2002) ont proposé un modèle proche du modèle de supercouverture mais basé sur une approche complexe cellulaire. Dans ce cas, la notion de *bulles* comme étant des *singularités* du modèle disparaît. Cependant, ce modèle n'est disponible qu'en dimensions 2 et 3 et n'est pas encore généralisé.

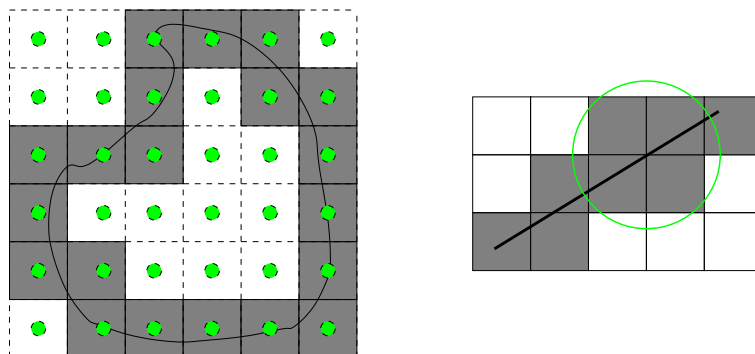


FIG. 1.11 –: *Discretisation par supercouvertures : (gauche) discretisation d'une courbe quelconque et illustration d'une bulle pour la discretisation d'une droite.*

1.6 Conclusion

Dans ce chapitre nous avons présenté quelques notions de base en géométrie discrète. Nous avons défini la notion d'espace discret, de point discret et d'objets simples sur cet espace (objet, courbe surface). Nous avons aussi présenté les différents codages des courbes dans le cas 2D. Nous avons terminé ce chapitre par une description des principales discrétisations d'objets euclidiens qui nous serviront par la suite.

Chapitre 2

Droites et Plans discrets

Sommaire

2.1	Introduction	20
2.2	Droites discrètes 2D	20
2.2.1	Définitions et propriétés	20
2.2.1.1	Caractérisation	21
2.2.1.2	Droites discrètes arithmétiques	23
2.2.1.3	Quelles <i>droites</i> choisir ?	28
2.2.1.4	Tracé de droites discrètes	29
2.2.2	Reconnaissance et segmentation	30
2.2.2.1	Approche basée sur la structure du dual	31
2.2.2.2	Approche arithmétique	38
2.2.2.3	Comparaisons entre approche arithmétique et approche duale	42
2.2.2.4	Bilan des algorithmes de reconnaissance présentés	44
2.2.2.5	De la reconnaissance à la segmentation de courbe	44
2.3	Droites discrètes 3D	44
2.3.1	Caractérisation et propriétés des droites discrètes 3D	46
2.3.2	Tracé de droites discrètes 3D	47
2.3.3	Reconnaissance et segmentation	51
2.4	Plans Discrets	54
2.4.1	Définitions et propriétés	54
2.4.2	Reconnaissance et facettisation d'objets discrets	61
2.4.2.1	Approches de KIM, VEELAERT et DEBLED-RENNESON	61
2.4.2.2	Approche basée sur la structure du dual	62
2.4.2.3	Vers une structure complète du domaine dans l'espace dual	64
2.4.2.4	Facettisation d'objets discrets	70
2.5	Statistique pour la reconnaissance de droites et plans discrets	73
2.6	Conclusion	78

2.1 Introduction

Dans ce chapitre nous nous intéressons aux objets fondamentaux de la géométrie discrète que sont les droites et les plans discrets. Ces objets de base permettent l'élaboration, sur le plan théorique, d'une géométrie discrète.

Dans les deux premiers paragraphes, nous analysons les différentes définitions et algorithmes associés aux droites discrètes dans \mathbb{Z}^2 et dans \mathbb{Z}^3 . Nous reprenons les différentes études présentées dans la littérature pour lesquelles nous proposons une évaluation comparative.

Dans le paragraphe 2.4, nous effectuons un travail similaire sur les plans discrets en proposant un premier théorème de structure de la *pré-image* d'un plan discret.

Enfin, dans le paragraphe 2.5, nous présentons une alternative à l'approche géométrique pour la reconnaissance de droites et plans discrets. Cette nouvelle approche se base sur une analyse statistique de ces objets.

2.2 Droites discrètes 2D

Nous commençons notre analyse des objets discrets par la notion de droite discrète. Les définitions, propriétés et algorithmes associés à ces objets sont primordiaux dans de nombreux domaines, par exemple en analyse d'image ou en reconnaissance de formes.

La notion de droite discrète est importante, non seulement sur le plan théorique comme objet de base du modèle discret, mais aussi cet objet et les algorithmes qui lui sont associés sont des outils de base pour des estimations de mesures euclidiennes sur des objets discrets. Ainsi, une bonne maîtrise de cette *brique de base* de la géométrie discrète est nécessaire avant de voir ses applications dans la caractérisation de formes des chapitres 4 et 5.

2.2.1 Définitions et propriétés

Sur la grille discrète, de nombreuses approches existent pour décider si un ensemble de pixels a une allure *rectiligne*. Bien évidemment, nous devons tout d'abord définir un peu plus formellement la notion de linéarité pour un ensemble de points discrets. Dans ce qui suit, nous utilisons les notions de *quadrant* ou *d'octant* pour un segment de droite en fonction d'une partition de l'espace 2D en 4 ou 8 régions (voir figure 2.1).

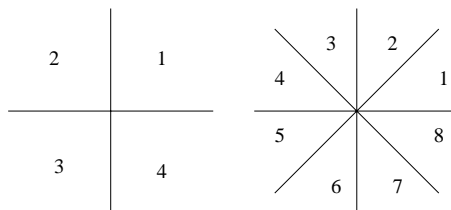


FIG. 2.1 –: *Décomposition de l'espace 2D en quadrants (gauche) et en octants (droite).*

D'une manière générale, un ensemble de points de la grille appartient à une *droite discrète* s'il existe une droite réelle dont la discrétisation contient les pixels considérés. Par exemple, si nous considérons le processus de discrétisation OBQ

(*Object Boundary Quantization*), un ensemble de pixels \mathcal{E} appartient à une droite discrète s'il existe au moins un couple (α, β) de \mathbb{R}^2 tel que tous les points (x, y) de \mathcal{E} vérifient :

$$0 \leq \alpha x + \beta - y < 1$$

Sur le plan méthodologique, certaines approches partent de ce processus de discrétisation pour énoncer les propriétés des droites discrètes qu'elles proposent ; d'autres ne travaillent qu'au niveau de la grille, elles énoncent les propriétés et ensuite montrent qu'une droite réelle discrétisée vérifie celles-ci.

Avant d'énoncer les constructions classiques de ces objets, la figure 2.2 illustre les subtilités, au moins sur le plan visuel, pour qu'un arc discret soit, ou non, un segment discret (figure tirée de VIALARD 1996a). Dans ce cas, la figure (a) est un morceau de droite discrète, (b) n'en est pas un et (c) non plus.

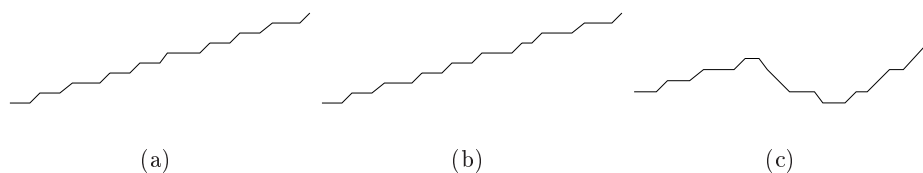


FIG. 2.2 –: *Seul l'arc (a) est un segment discret selon les définitions usuelles des droites discrètes (cf ci-dessous).*

2.2.1.1 Caractérisation

Considérons tout d'abord le cas d'un arc discret 8-connexe. Une des premières caractérisations des droites discrètes a été proposée par FREEMAN (1974) et est basée sur le codage de l'arc. Ainsi, un arc est un segment discret si :

- son codage ne contient que deux codes différents et ceux-ci ne diffèrent que de 1 (modulo 8) ;
- un de ces deux codes est toujours isolé dans le codage ;
- ce code isolé apparaît dans le codage le plus uniformément possible.

Dans cette caractérisation, la dernière propriété est assez floue et cette définition globale d'un segment discret n'est pas encore satisfaisante.

Par la suite, ROSENFELD (1974) introduit une propriété importante permettant de caractériser un segment discret : *la propriété de la corde*. Tout ensemble de points discrets qui vérifie cette propriété est un morceau de droite discrète et réciproquement, tous les points d'un segment discret vérifient celle-ci.

Cette propriété s'énonce de la façon suivante (voir figure 2.4) :

Définition 2.1 (Propriété de corde) *Un ensemble de pixels \mathcal{E} vérifie la propriété de corde si pour tout couple p et q de \mathcal{E} , et pour tout point $m(x, y)$ du segment $[pq]$, il existe un point $M(i, j)$ de \mathcal{E} tel que*

$$\max(|i - x|, |j - y|) < 1$$

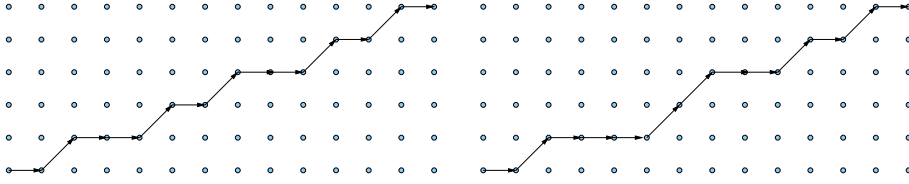


FIG. 2.3 – : *Propriété de FREEMAN* : (gauche) le codage est $0 - 1 - 0 - 0 - 1 - 0 - 1 - 0 - 0 - 1 - 0 - 1 - 0$, les codes “1” apparaissent isolés et uniformément répartis ; (droite), le codage est $0 - 1 - 0 - 0 - 0 - 1 - 1 - 0 - 0 - 1 - 0 - 1 - 0$, dans ce cas, ni le code “0” ni le code “1” sont isolés, cet arc n’est pas un segment discret.

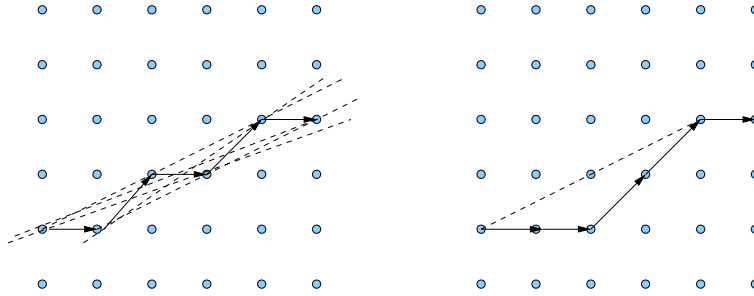


FIG. 2.4 – : *Propriété de corde de ROSENFELD* : (gauche) tous les segments $[pq]$ de longueur supérieure à 2 vérifient cette propriété, cet arc est donc un segment discret ; (droite), existence d’un contre-exemple, ce segment ne correspond donc pas à un morceau de droite discrète.

Cette propriété permet une caractérisation formelle des segments de droites discrètes mais n’est pas constructive dans le sens où elle ne permet pas directement de tracer ou de reconnaître un tel segment autrement que par un test exhaustif.

En se basant sur une réécriture des critères de FREEMAN par HÜBLER ET AL. (1981), WU (1982) prouve l’équivalence entre ces critères et la propriété de corde de ROSENFELD.

Dans une approche similaire à la propriété de corde, HUNG (1985) propose une caractérisation des droites discrètes basée sur la notion d’uniformité ou de régularité (traduction de *evenness*, voir figure 2.5).

Définition 2.2 (Propriété de régularité) *Un ensemble de pixels \mathcal{E} dans le premier octant est dit régulier si tout quadruplet de points (a, b, c, d) de \mathcal{E} , tel que $\vec{ba}_x = \vec{dc}_x$, vérifie :*

$$|\vec{ba}_y - \vec{dc}_y| = 1$$

DORST et SMEULDERS (1984) proposent non seulement une caractérisation des droites discrètes, mais aussi un codage unique des segments discrets. A partir d’un codage de FREEMAN d’un segment discret, ils introduisent un quadruplet d’entiers (n, q, p, s) décrivant entièrement ce segment discret. Considérons, sans perte de généralités, un segment dans le premier octant. Le codage de FREEMAN est donné par les codes $\{c_i\}_{i=1\dots n}$, nous avons alors (voir figure 2.6) :

- n représente la longueur du segment de droite discrète ;

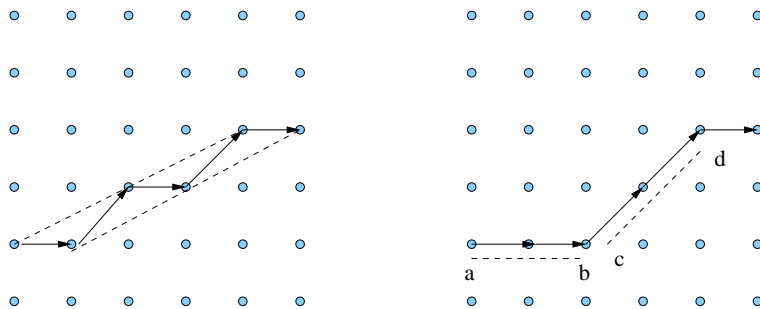


FIG. 2.5 –: *Propriété de régularité de HUNG : (gauche) tout quadruplet (a, b, c, d) vérifie la propriété de régularité, cet arc est donc un segment discret ; (droite), existence d'un contre-exemple, les vecteurs \vec{ab} et \vec{cd} ne vérifient pas cette propriété, ce segment ne correspond donc pas à un morceau de droite discrète.*

- q la période dans le code de FREEMAN. Ce paramètre s'écrit formellement :

$$q = \min_k \{k \in \{1, 2, \dots, n\} \mid k = n \vee \forall i \in \{1, 2, \dots, n - k\} \quad c_{i+k} = c_i\}$$

- p est la hauteur sur la période, c'est-à-dire :

$$p = \sum_1^q c_i$$

- s est le déphasage du segment par rapport à la période q . Nous avons formellement :

$$s : s \in \{0, 1, \dots, q-1\} \mid \forall i \in \{1, 2, \dots, q\} : c_i = \left\lfloor \frac{p}{q}(i - s) \right\rfloor - \left\lfloor \frac{p}{q}(i - s - 1) \right\rfloor$$

D'une manière informelle, le paramètre q correspond à la règle *d'uniformité* de FREEMAN (troisième critère).

A partir de ce codage (n, q, p, s) , une droite euclidienne telle que sa discrétisation avec OBQ correspond au segment discret est donnée par :

$$y = \frac{p}{q}(x - s) + \left\lfloor \frac{sp}{q} \right\rfloor$$

Cependant, l'ensemble des droites solutions se caractérise entièrement avec ce quadruplet. Ainsi, DORST et SMEULDERS (1984) détaillent la construction de ce domaine des droites ou *pré-image* dans l'espace des paramètres (α, β) des pentes et des ordonnées à l'origine des droites.

Dans une même approche de construction du domaine des droites ou de la *pré-image* de l'arc discret, LINDENBAUM et KOPLOWITZ (1991) proposent un autre codage basé sur les séries de FAREY (voir paragraphe 2.2.2.1) et décrivent un processus de conversion entre ce codage et celui de DORST et SMEULDERS. Nous reviendrons dans le paragraphe suivant sur les différentes techniques de reconnaissance de droites discrètes basées sur ce type de codage.

2.2.1.2 Droites discrètes arithmétiques

Nous présentons une dernière caractérisation que nous allons beaucoup utiliser par la suite, celle proposée par REVEILLÈS (1991) :

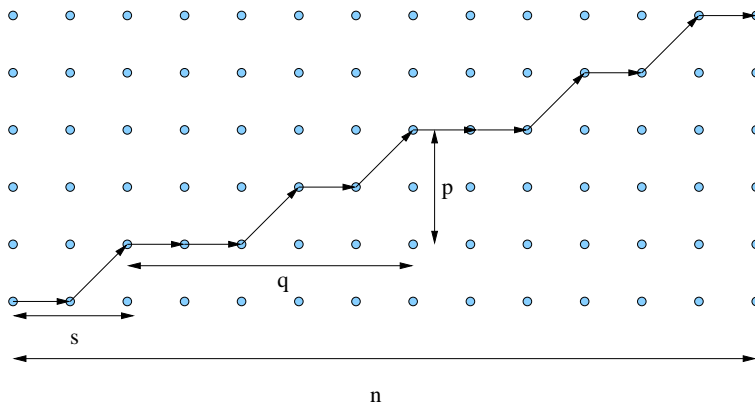


FIG. 2.6 – : Représentation graphique des quadruplets (n, q, p, s) de DORST et SMEULDERS pour la caractérisation des droites discrètes ; notre segment se code donc $(14, 5, 2, 2)$.

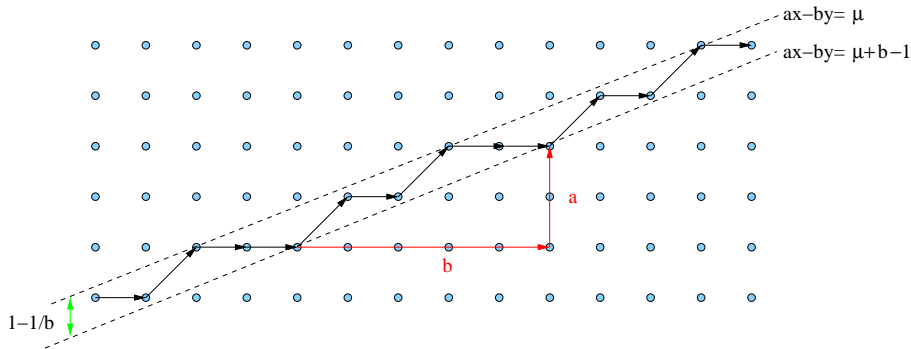


FIG. 2.7 – : Droite discrète arithmétique naïve de paramètres $\mathcal{D}(2, 5, -1, 5)$.

Définition 2.3 (Droite discrète arithmétique) Un ensemble de pixels \mathcal{E} appartient à la droite discrète arithmétique de pente $\frac{a}{b}$, de borne inférieure μ et d'épaisseur ω (avec a, b, μ et ω dans \mathbb{Z} , $b \neq 0$ et $\text{pgcd}(a, b) = 1$), si et seulement si tous les pixels (x, y) de \mathcal{E} vérifient :

$$\mu \leq ax - by < \mu + \omega \tag{2.1}$$

Cette droite se note $\mathcal{D}(a, b, \mu, \omega)$.

D'un point de vue géométrique, cette droite discrète peut se construire en considérant l'ensemble des points de la grille contenus dans une bande définie par les droites (voir figure 2.7) :

$$\begin{cases} ax - by = \mu \\ ax - by = \mu + \omega \quad (\text{exclue}) \end{cases}$$

Cette caractérisation basée sur la notion de *bande* a aussi été utilisée par KOVALEVSKY (1990) pour définir les droites discrètes.

Dans cette approche, le paramétrage de la droite par des entiers avec la propriété que a et b soient premiers entre eux, nous permet d'intégrer, à cette notion géométrique, des propriétés provenant de l'arithmétique ou de la théorie des nombres.

Commençons par lier cette notion aux droites discrètes classiques. Pour cela, nous avons le théorème de connexité des droites discrètes suivant (REVEILLÈS, 1991) :

Théorème 2.1 (connexité des droites discrètes) *Étant donnée une droite discrète $\mathcal{D}(a, b, \mu, \omega)$ alors :*

- si $\omega < \max(|a|, |b|)$ la droite discrète est déconnectée ;
- si $\omega = \max(|a|, |b|)$ la droite discrète est 8-connexe (ou est un 8-arc), on parle alors de droite **naïve** ;
- si $\max(|a|, |b|) < \omega < |a| + |b|$, la droite est ***-connexe**, c'est-à-dire qu'elle présente des 4- et des 8-connexités ;
- si $\omega = |a| + |b|$ la droite est strictement 4-connexe (ou est un 4-arc), on parle de droite **standard** ;
- si $\omega > |a| + |b|$ on parle de **droite épaisse**.

Ces connexités sont illustrées figure 2.8.

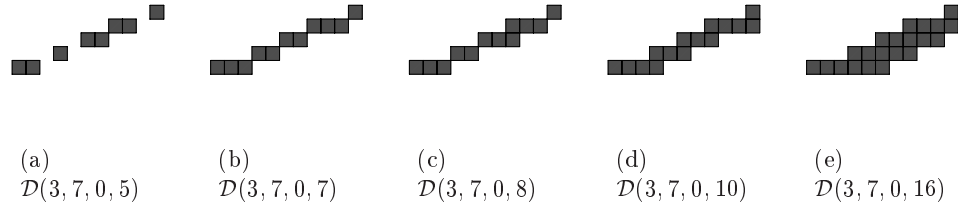


FIG. 2.8 — Connexité des droites discrètes en fonction de ω : (a) segment déconnecté, (b) segment naïf, (c) segment *-connexe, (d) segment standard et (e) segment épais.

Nous pouvons faire le lien entre ces droites et les processus de discrétisation usuels :

Proposition 2.1 (REVEILLÈS 1991) *Supposons une droite rationnelle D d'équation $ax + by + c = 0$ avec a, b, c dans \mathbb{Z} , on suppose sans perte de généralité que $b = |b| = \max(|a|, |b|)$. Nous avons alors :*

- la discrétisation par défaut de D , c'est-à-dire l'ensemble des pixels tel que $\{(x, y) \mid y = \lfloor \frac{-ax-c}{b} \rfloor\}$ coïncide avec la droite discrète $\mathcal{D}(a, b, -b - c + 1, b)$;
- la discrétisation par excès de D , c'est-à-dire l'ensemble des pixels tel que $\{(x, y) \mid y = \lceil \frac{-ax-c}{b} \rceil\}$ coïncide avec la droite discrète $\mathcal{D}(a, b, -c + 1, b)$;
- la discrétisation de meilleure approximation (ou GIQ) de D , c'est-à-dire l'ensemble des pixels tel que $\{(x, y) \mid y = \lfloor \frac{-ax-c}{b} \rfloor\}$ coïncide avec la droite discrète $\mathcal{D}(a, b, -c + 1 - \lfloor \frac{b}{2} \rfloor, b)$.

Revenons sur la construction des droites arithmétiques. Avant de détailler les éléments qui vont nous servir dans les processus de reconnaissance, il nous faut définir les structures qui nous seront utiles.

Ainsi, étant donnés a, b, μ et ω dans \mathbb{Z} avec $\text{pgcd}(a, b) = 1$, on appellera **pointillé** de niveau k l'ensemble des pixels (x, y) vérifiant l'équation diophantienne :

$$ax - by = k$$

La structure d'un pointillé est assez simple, si le point (x_0, y_0) appartient au pointillé de niveau k , tous les autres points sont de la forme :

$$(x, u) = (x_0, y_0) + k'(b, a)$$

avec k' dans \mathbb{N} . Au niveau des pixels composant ce pointillé, on définit le **reste** r d'un pixel (x, y) selon le couple (a, b) comme étant un point du pointillé de niveau r . Le terme **reste** vient de la notion usuelle de reste d'une division entière. En effet, nous avons :

$$r = \left\{ \frac{ax}{b} \right\}$$

Nous avons maintenant la propriété suivante nous permettant de construire des droites discrètes à partir de pointillés. Les deux items de cette propriété sont identiques par définition mais ces deux écritures différentes nous serons utiles par la suite :

Proposition 2.2 *La droite $\mathcal{D}(a, b, \mu, \omega)$ est :*

- la réunion des pointillés de niveau k pour k dans $[\mu, \mu + \omega[$
- l'ensemble des pixels de reste compris entre $[\mu, \mu + \omega[$

Sur cette droite discrète, nous pouvons encore remarquer des objets importants. Ainsi, on appellera **droite d'appui inférieure** de \mathcal{D} , la droite d'équation :

$$ax - by = \mu + \omega - 1$$

Les pixels de cette droite (*i.e* les pixels de reste $\mu + \omega - 1$) sont appelés **points d'appui inférieurs**.

De manière identique, on appellera **droite d'appui supérieure** de \mathcal{D} , la droite d'équation :

$$ax - by = \mu$$

Les pixels de cette droite (*i.e* les pixels de reste μ) sont appelés **points d'appui supérieurs** (voir figure 2.9).

Certaines notions d'arithmétique liées à ces droites apparaissent très nettement dans la définition des restes associés à une droite discrète. On se place en effet dans un espace modulaire *modulo* b , ce qui illustre la notion de *période* b associée à une droite discrète (qui correspond exactement au paramètre q de DORST et SMEULDERS 1984).

Ces notions de modularité dans les droites discrètes sont très anciennes ; en effet BERNOULLI (1771), qui était *astronome royal*, présente un chapitre dont le titre est "*Sur une nouvelle espèce de Calcul*". Le problème de l'époque était la construction d'éphémérides pour l'astronomie. BERNOULLI présente une méthode très efficace pour l'interpolation linéaire de la position de planètes. En se ramenant à des positions entières, l'objectif est de calculer les valeurs de P en fonction de n dans l'expression $P = \frac{mn}{c}$.

[à propos du calcul d'interpolation classique (BERNOULLI, 1771)] *or la Méthode que je me propose de rendre compte, satisfait exactement cette condition, & quel que soit le nombre de n , on n'a besoin de faire, pour une valeur donnée de m , qu'une seule règle de trois, tous les autres termes se déterminant par de simples additions, au moyen de certaines formules*

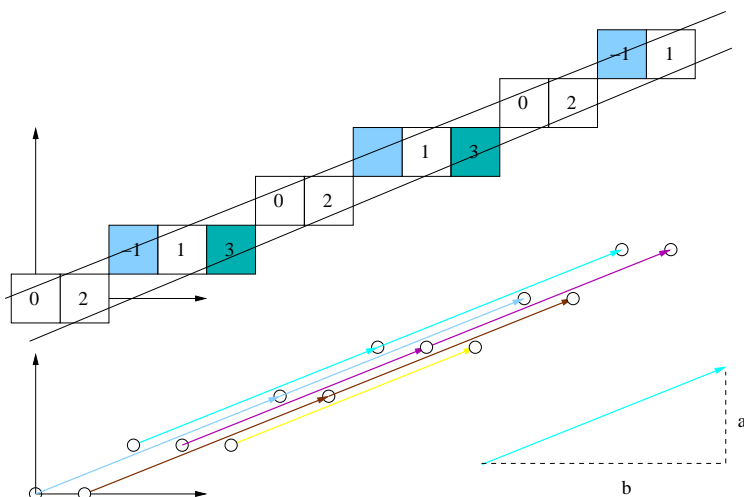


FIG. 2.9 — : Droite discrète arithmétique naïve de paramètres $\mathcal{D}(2, 5, -1, 5)$: (haut) représentation par restes avec les points et droites d'appui (bas) construction par union de pointillés.

générales que j'ai trouvées ; Et cette méthode est d'une telle facilité dans l'application, qu'on est souvent en état d'écrire, sans autre calcul, les produits de plus de 1000 règles de trois, en deux heures de temps.

Il émet ensuite l'observation suivante :

On peut remarquer que la différence pour ces augmentations des P ne roulera jamais que sur une unité de plus ou de moins ; c'est à dire que si $m = x + \frac{a}{c}$, ce sera toujours ou x ou $x + I$ unités qu'il faudra ajouter au dernier chiffre à mesure que n croît de I .

En d'autres termes, si n et P sont respectivement des abscisses et des ordonnées, les pixels successifs pour x croissant, appartenant à la droite décrite par la pente $\frac{m}{c}$, sont constitués en *palier* avec des décrochements diagonaux entre ceux-ci.

Il montre ensuite l'aspect périodique de ces décrochements :

De plus, si on entend par $\frac{h}{k}$ la fraction $\frac{a}{c}$ réduite à ses moindres termes, ou que $\frac{m}{c} = x + \frac{a}{c} = x + \frac{h}{k}$; il ne m'a pas été difficile de prouver, qu'au bout d'une période de k termes les augmentations de P doivent revenir dans le même ordre ; c'est à dire que si on nomme Π le terme correspondant à $n + k$, il faudra ajouter à Π , pour $n + k + I$, le même nombre d'unités qu'on avait ajouté à P pour $n + I$ [...].

Il propose ensuite un calcul basé sur la fraction continue de $\frac{h}{k}$ pour construire explicitement tous les pixels de la droite discrète (voir HARDY et WRIGHT 1975 pour un complément sur ces notions d'arithmétique et de théorie des nombres). Ce premier travail a ensuite été repris par CHRISTOFFEL (1875) qui construit un mot qui, par une transformation simple, se ramène à une droite discrète arithmétique (REVEILLÈS, 1991).

Après cette parenthèse historique, revenons sur la construction de ces objets. Nous illustrons la puissance de ce formalisme en donnant une interprétation gra-

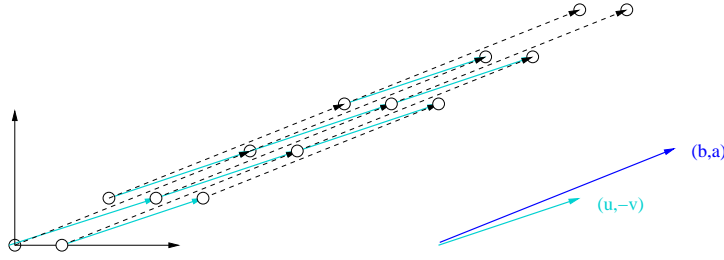


FIG. 2.10 – Pointillés et vecteur de Bezout pour la droite $\mathcal{D}(2, 5, -1, 5)$.

phique du théorème de BEZOUT, très classique en arithmétique :

$$\forall a, b \in \mathbb{N}, \quad \exists u, v \in \mathbb{N} : au + bv = \text{pgcd}(a, b)$$

Dans notre analyse, a et b sont des premiers entre eux, on considère u et v les coefficients de BEZOUT donnés par:

$$au - bv = 1$$

Ainsi, si (x_0, y_0) appartient au pointillé de niveau k , les autres points de même reste sont générés par application du vecteur $(b, a)^T$. Un *vecteur de Bezout* $(u, v)^T$ nous permet de passer d'un pointillé à un autre. En effet, tous les vecteurs de la forme $(u, v)^T + k'(b, a)^T$ avec k' dans \mathbb{Z} nous permettent de passer d'un point du pointillé k à un point du pointillé $k + 1$. Pour cela, supposons $ax_0 - by_0 = k$ puis :

$$\begin{aligned} a(x_0 + u + k'b) - b(y_0 + v + k'a) &= ax_0 - by_0 + au - bv + abk' - abk' \\ &= ax_0 - by_0 + 1 \\ &= k + 1 \end{aligned}$$

Par exemple, la figure 2.10 illustre les vecteurs générant la droite $\mathcal{D}(2, 5, -1, 5)$ tels que $(b, a)^T = (5, 2)^T$ et $(u, v)^T = (3, 1)^T$. Ainsi, la construction d'une droite discrète se fait, en partant d'un point de reste μ , par application de $\omega - 1$ fois le vecteur de Bezout pour passer d'un pointillé à l'autre et autant de fois le vecteur (b, a) qu'il faut dans chaque pointillé. Nous utiliserons cette notion de vecteur de Bezout dans la partie reconnaissance de cercles (chapitre 3).

2.2.1.3 Quelles droites choisir ?

De nombreuses autres définitions de droites discrètes existent, le lecteur pourra se référer à l'article de ROSENFELD et KLETTE (2001) ou aux thèses soutenues sur ce sujet comme celles de DEBLED-RENNESON (1995) ou de VIALARD (1996a). Cependant, la majorité de ces définitions désigne le même objet géométrique. Dans ce que nous avons présenté, nous avons l'équivalence entre les définitions suivantes :

- les propriétés de FREEMAN formalisées par WU et HÜBLER ET AL. ;
- la propriété de corde de ROSENFELD ;
- la propriété de régularité de HUNG ;
- le quadruplet (n, q, p, s) de DORST et SMEULDERS ;

- la notion de droite arithmétique avec comme épaisseur $\omega = \max(|a|, |b|)$.

Le choix de telle ou telle approche est dirigé par l'algorithmique que l'on applique à ces définitions. Par exemple, si notre objectif est le tracé de droite, la propriété de corde ne nous permet pas d'écrire des algorithmes efficaces alors que les droites arithmétiques le permettent. De même, dans un objectif de reconnaissance de droite, on adoptera la méthode la plus simple à implémenter.

2.2.1.4 Tracé de droites discrètes

Nous présentons les algorithmes classiques de tracé de droites discrètes que nous utiliserons par la suite.

Nous commençons par un algorithme très connu proposé par BRESENHAM (1965). Le cadre de ce travail est le suivant : nous considérons deux points de la grille notés (x_0, y_0) et (x_1, y_1) et l'on souhaite tracer le segment de droite joignant ces deux pixels. Nous pouvons, sans perte de généralité, nous placer dans le premier octant, c'est-à-dire $x_1 - x_0 > y_1 - y_0 > 0$. Nous illustrerons par la suite les transformations permettant de généraliser le tracé dans les autres octants.

L'idée de cet algorithme est le suivant : nous allons tracer les pixels de x_i vers x_f dans cet ordre. A chaque étape, nous devons choisir entre rester sur le même *palier* (*i.e.* rester à la même ordonnée) ou aller en diagonale. Pour cela, nous propageons une mesure d'erreur nous permettant de toujours choisir le pixel le plus proche. Nous obtenons donc l'algorithme 1.

Algorithme 1 Tracé de droite de BRESENHAM (1965)

```

TRACÉ_BRESENHAM( $x_0, y_0, x_1, y_1$ )
1:  $dx = x_1 - x_0, dy = y_1 - y_0$ 
2:  $x = x_0, y = y_0$ 
3:  $incrHor = 2dy, incrDiag = 2(dx - dy)$ 
4:  $e = 2dy - dx$ 
5: pour  $x$  allant de  $x_0$  à  $x_1$  faire
6:   AFFICHER_PIXEL( $x, y$ )
7:   si  $e \geq 0$  alors
8:      $y += 1$ 
9:      $e += incrDiag$ 
10:  sinon
11:     $e += incrHor$ 
12:  fin du si
13: fin du pour

```

Nous détaillons aussi l'algorithme de tracé des droites discrètes arithmétiques. Nous nous intéressons plus particulièrement aux droites arithmétiques naïves. Comme nous l'avons remarqué précédemment, la structure modulaire du reste dans la droite discrète permet de comprendre entièrement la structure géométrique de la droite. Il est donc naturel de baser l'algorithme de tracé sur cette information. Nous obtenons donc l'algorithme 2. Cet algorithme est très similaire à celui de BRESENHAM, en fait, l'algorithme 1 est une instance de cet algorithme pour $\mu = \left\lceil \frac{x_f - x_i}{2} \right\rceil + 1$ (voir DEBLED-RENNESON 1995).

D'autres algorithmes de tracé pour les droites arithmétiques *épaisses* ont été proposés par REVEILLÈS (1991) mais nous ne les utiliserons pas par la suite.

Algorithme 2 Tracé de droites arithmétiques naïves de REVEILLÈS (1991) et DEBLED-RENNESON (1995)

TRACÉ_ARITHMÉTIQUE(x_0, y_0, x_1, y_1, μ)

```

1:  $v_x = x_1 - x_0, v_y = y_1 - y_0$ 
2:  $r = v_y x_0 - v_x y_0 - \mu$ 
3:  $x = x_0, y = y_0$ 
4: tant que  $x < x_b$  faire
5:    $x += 1$ 
6:    $r += v_y$ 
7:   si  $r < 0$  ou  $r \geq v_x$  alors
8:      $y += 1$ 
9:      $r -= v_x$ 
10:  fin du si
11:  AFFICHER_PIXEL( $x, y$ )
12: fin du tant que

```

Pour généraliser ce tracé dans tous les octants, nous utilisons, au niveau de la fonction AFFICHE_PIXEL, les transformations du tableau 2.1. Le nombre de tests à effectuer dans l'implémentation est encore suffisamment raisonnable pour envisager une implémentation directe du tableau 2.1. Nous verrons par la suite comment nous résolvons le problème en 3D où 48 cas sont à gérer.

Octants	Transformations de (x, y)
1	(x, y)
2	(y, x)
3	$(y, -x)$
4	$(x, -y)$
5	$(-x, -y)$
6	$(-y, -x)$
7	$(-y, x)$
8	$(-x, y)$

TAB. 2.1 –: Transformations pour la généralisation du tracé de droite dans tous les octants.

2.2.2 Reconnaissance et segmentation

Dans cette partie, nous nous intéressons aux algorithmes permettant de décider si un ensemble de points discrets appartient ou non à une droite discrète. De manière plus précise, nous pouvons décomposer le problème en deux grands axes :

Test de linéarité : prédicat permettant de décider si un ensemble de pixels forme un segment discret.

Reconnaissance de la droite : cette fois, non seulement nous décidons si un ensemble de pixels appartient à une droite discrète, mais nous obtenons les paramètres de cette droite discrète.

Quel que soit le problème choisi, nous évaluons les différents algorithmes selon les critères suivants :

Conditions sur l'ensemble de pixels : est-ce que l'algorithme émet des hypothèses sur la structure de l'ensemble de pixels (connexité, tri sur un axe de coordonnées...) ?

Complexité : mesurée en fonction du nombre de pixels de l'ensemble à reconnaître. Nous nous intéressons aussi à la complexité en mémoire de chaque algorithme.

Difficulté d'implémentation : d'un point de vue pratique, est-ce que la méthode se programme facilement ?

Dans la littérature, ROSENFELD et KLETTE (2001) attribuent le premier algorithme de reconnaissance de droite discrète en temps linéaire à HÜBLER ET AL. (1981). Cependant, le problème peut être abordé par tellement d'approches différentes que cela rend toute classification historique difficile.

Dans ce qui suit, nous détaillerons deux grandes approches pour la reconnaissance qui seront utilisées par la suite. Le lecteur pourra se référer à ROSENFELD et KLETTE (2001) pour une bibliographie plus large sur le sujet.

2.2.2.1 Approche basée sur la structure du dual

A partir du codage (n, q, p, s) des droites discrètes, DORST et SMEULDERS (1984) construisent la *pré-image* associée à une droite discrète. Cette pré-image est un domaine, dans l'espace des paramètres, qui représente l'ensemble des droites réelles dont la discrétisation contient l'ensemble des pixels du segment discret considéré.

De manière plus formelle, nous considérons tout d'abord un segment de droite discrète naïve dans le premier octant, noté \mathcal{S} . D'après les définitions vues au paragraphe 2.2.1, il existe alors un couple $(\alpha, \beta) \in [0, 1] \times [0, 1[$ tel que \mathcal{S} est contenu dans la discrétisation de la droite $y = \alpha x + \beta$.

Nous notons $\bar{\mathcal{S}}$ l'ensemble de ces droites donné par :

$$\bar{\mathcal{S}} = \{(\alpha, \beta) \in [0, 1] \times [0, 1[\mid \forall (x, y) \in \mathcal{S}, 0 \leq \alpha x + \beta - y < 1\} \quad (2.2)$$

Si nous décomposons la construction de $\bar{\mathcal{S}}$, chaque pixel de \mathcal{S} engendre, dans l'espace dual (α, β) un bande semi-ouverte définie par :

$$\mathcal{B}(x, y) = \begin{cases} \alpha x + \beta - y \geq 0 \\ \alpha x + \beta - y < 1 \end{cases}$$

Étant donné que $\bar{\mathcal{S}}$ est une intersection de contraintes linéaires, ce domaine est convexe dans l'espace des paramètres (voir figure 2.11).

Présenté en ces termes, le problème se ramène donc à un problème classique de programmation linéaire : étant donné un système de contraintes linéaires, nous voulons soit décider si le système est valide (c'est-à-dire s'il existe au moins une solution), soit calculer le polytope solution dans l'espace des paramètres.

De nombreux algorithmes existent pour résoudre ce problème quelque soit la dimension des contraintes (c'est-à-dire la dimension de l'espace des paramètres). De manière générale, nous avons le théorème suivant nous permettant d'avoir une borne sur la complexité asymptotique du test de linéarité :

Théorème 2.2 (MEGIDDO (1984)) *Étant donné un système de n inéquations linéaires de dimension d , tester si un système admet une solution se calcule en temps optimal pour une dimension fixe, c'est-à-dire en $O(n)$.*

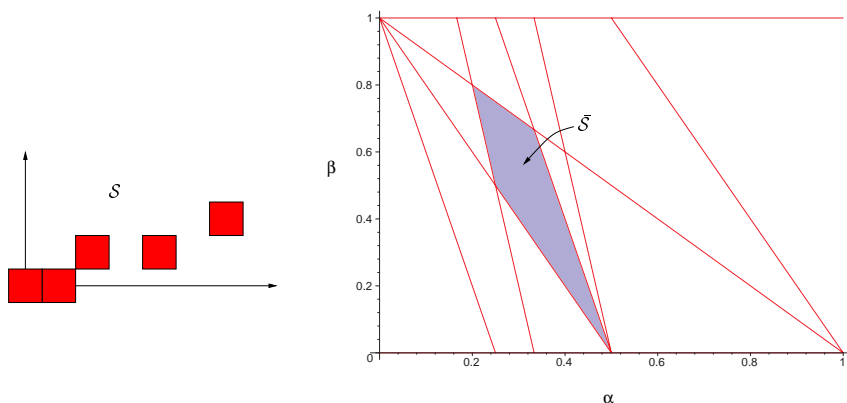


FIG. 2.11 –: Ensemble de pixels S et domaine des droites euclidiennes $y = \alpha x + \beta$ associées, noté \bar{S} .

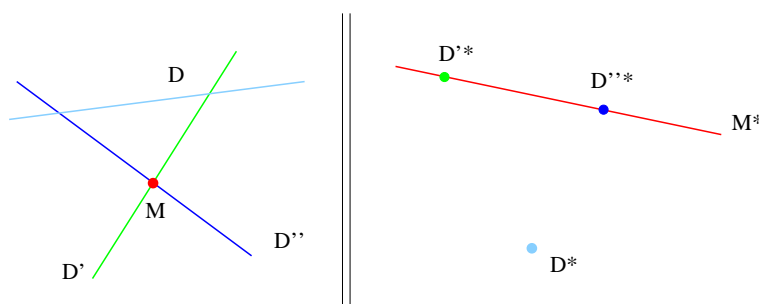


FIG. 2.12 –: Passage dans l'espace des paramètres associés aux droites : les droites D , D' et D'' (à gauche) deviennent respectivement les points D^* , D'^* et D''^* (à droite), le point M devient la droite M^* . M satisfait la contrainte linéaire D implique que D^* est en dessous de M^* dans l'espace dual.

Le problème majeur de cette approche est la constante importante devant cette complexité asymptotique. En effet, celle-ci est exponentielle en la dimension. Cependant, dans le cas de faible dimension, l'algorithme de MEGIDDO (1984) offre une solution très efficace. Dans notre cas, le test de linéarité d'un ensemble de pixels engendre un système de $2n$ inéquations de dimension 2 (où n est le nombre de pixels). Nous obtenons donc une solution optimale en temps, c'est-à-dire $O(n)$, à notre premier problème. Récemment, BUZER (2002) a présenté une version incrémentale de cet algorithme toujours en temps optimal quelle que soit la dimension.

Si nous souhaitons décrire tous les sommets du domaine solution, des algorithmes très efficaces existent pour les dimensions 2 et 3. Ces algorithmes viennent généralement d'une résolution du problème dual : en dimension 2, la transformée dans l'espace des paramètres d'une droite est un point, la transformée d'un point est une droite et une inéquation linéaire se transforme aussi en une contrainte linéaire dans le dual mais calculer le polytope des solutions d'un système d'inéquation se ramène à un calcul d'enveloppe convexe dans le dual (voir BOISSONNAT et YVINEC 1995; DE BERG ET AL. 2000 et figure 2.12).

Ainsi, les bornes des algorithmes de calcul d'enveloppes convexes impliquent des bornes sur le calcul du domaine des solutions d'un système. Pour la dimension 2,

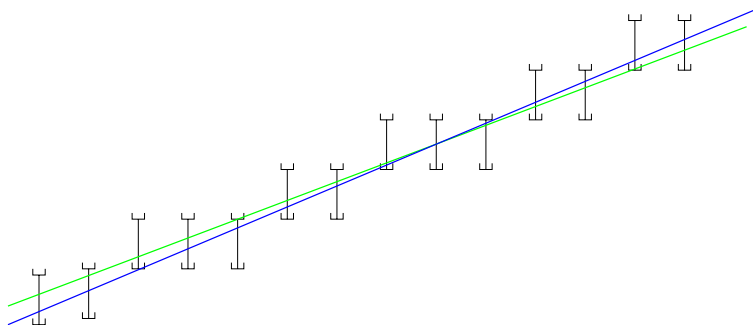


FIG. 2.13 –: Représentation de la droite discrète $\mathcal{D}(2, 5, -1, 5)$ sous forme d'intervalles et droites euclidiennes solutions.

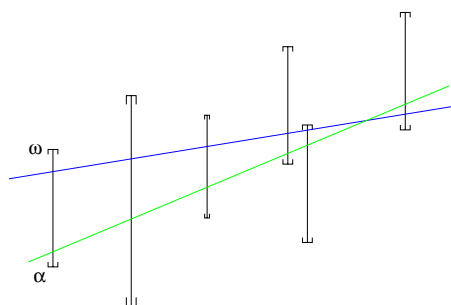


FIG. 2.14 –: Algorithme de O'ROURKE pour construire l'ensemble des droites réelles passant par une série d'intervalles.

PREPARATA et SHAMOS (1985) ont présenté un algorithme optimal dont la borne est en $O(n \log(n))$. Pour ne pas surcharger ce chapitre, l'algorithme optimal de PREPARATA et SHAMOS (1985) est présenté en annexe (voir annexe C).

Finalement, sans aucune prise en compte de l'aspect géométrique ou des propriétés des droites discrètes, nous avons des algorithmes très efficaces pour le test de linéarité ou la reconnaissance de droites discrètes. Ces bornes sont certes intéressantes mais l'introduction de la particularité des droites discrètes permet de simplifier considérablement les bornes et les algorithmes.

Avant de présenter un premier algorithme de reconnaissance optimal, il nous faut revenir sur une illustration géométrique de l'équation 2.2. Ainsi, l'ensemble des solutions $\bar{\mathcal{S}}$ correspond à l'ensemble des droites passant par tous les segments semi-ouverts $[MM']$ où $M = (x, y)$ et $M' = (x, y + 1)$ pour tous les points (x, y) de \mathcal{S} (voir figure 2.13). Cette réécriture du problème est évidente mais elle nous permet d'appliquer un algorithme très efficace proposé par O'ROURKE (1981) permettant de décrire l'ensemble des droites réelles passant par un ensemble d'intervalles définis sous la forme $[(x, \alpha), (x, \omega)]$ (voir figure 2.14)

Ainsi, O'ROURKE présente un algorithme incrémental et optimal en temps pour décrire le domaine de l'ensemble des droites passant par ces intervalles (voir figure 2.14) : étant donné un ensemble de n intervalles triés selon l'axe des abscisses, le calcul du domaine des droites dans l'espace des paramètres se calcule en $O(n)$. De plus, pour tout ajout d'intervalle croissant en x , la mise à jour du domaine se fait au pire cas en $O(n)$ mais l'analyse amortie de ce calcul montre une complexité optimale pour le problème.

En considérant donc une description de \mathcal{S} sous forme d'une liste chaînée de pixels

(description en k -arcs), nous avons ramené la complexité de $O(n \log(n))$ à $O(n)$.

En ajoutant la contrainte de connexité, nous obtenons une borne optimale en temps et en mémoire à nos deux problèmes. Ainsi, lors de la caractérisation en (n, q, p, s) , DORST et SMEULDERS (1984) ont montré le théorème suivant :

Théorème 2.3 (Structure du domaine des droites discrètes) *Étant donné un ensemble de pixels \mathcal{S} connexes, le domaine $\bar{\mathcal{S}}$ dans l'espace des paramètres a , au plus, 4 sommets.*

La preuve de ce théorème a été présentée par DORST et SMEULDERS, et indépendamment, MCILROY (1985) a proposé, la même année, une preuve très simple concernant la propriété de structure du dual.

En plus de ce résultat sur le nombre de sommets dans le dual, ces sommets ont une structure arithmétique très intéressante. Tout d'abord, rappelons les notions de séries de Farey en théorie des nombres (voir par exemple HARDY et WRIGHT 1975) :

Définition 2.4 (Serie de Farey) *Une série de Farey d'ordre $n \in \mathbb{N}^*$ est l'ensemble des fractions irréductibles ordonnées de l'intervalle $[0, 1]$ dont les dénominateurs sont inférieurs à n .*

Par exemple, la série de Farey d'ordre 5 est l'ensemble :

$$\mathcal{F}_5 = \left\{ \frac{0}{1}, \frac{1}{5}, \frac{1}{4}, \frac{1}{3}, \frac{2}{5}, \frac{1}{2}, \frac{3}{5}, \frac{2}{3}, \frac{3}{4}, \frac{4}{5}, \frac{1}{1} \right\}$$

Sur ces séries, nous avons les propriétés suivantes :

Proposition 2.3 *Étant donnée \mathcal{F}_n la série de Farey d'ordre n , nous avons :*

- si $\frac{h}{k}$ et $\frac{h'}{k'}$ sont deux membres consécutifs de \mathcal{F}_n , alors $kh' - hk' = 1$
- si $\frac{h}{k}$, $\frac{h''}{k''}$ et $\frac{h'}{k'}$ sont trois termes consécutifs de \mathcal{F}_n , alors $\frac{h''}{k''} = \frac{h+h'}{k+k'}$. Ce point est appelé le **médian** de $\frac{h}{k}$ et $\frac{h'}{k'}$.

Les séries de Farey peuvent se calculer de manière récursive. La série \mathcal{F}_{n+1} se construit, à partir de la série \mathcal{F}_n , en ajoutant tous les points médians des fractions consécutives de \mathcal{F}_n dont les dénominateurs sont inférieurs à $n + 1$.

Ainsi, pour construire la série d'ordre 6, nous ajoutons à \mathcal{F}_5 les fractions $\frac{1}{6}$ et $\frac{5}{6}$ (tous les autres médians sont déjà dans la liste). Nous obtenons donc :

$$\mathcal{F}_6 = \left\{ \frac{0}{1}, \frac{1}{6}, \frac{1}{5}, \frac{1}{4}, \frac{1}{3}, \frac{2}{5}, \frac{1}{2}, \frac{3}{5}, \frac{2}{3}, \frac{3}{4}, \frac{4}{5}, \frac{5}{6}, \frac{1}{1} \right\}$$

En se basant sur ces définitions et propriétés des séries de Farey, nous pouvons compléter le théorème 2.3 en spécifiant encore la structure de l'espace dual associé aux droites discrètes :

Théorème 2.4 (MCILROY (1985); DORST et SMEULDERS (1984)) *Étant donné un ensemble \mathcal{S} de $N + 1$ pixels connexes dont l'abscisse minimale des points est x_0 , le domaine $\bar{\mathcal{S}}$ dans l'espace des paramètres est tel que :*

- il a au plus 4 sommets ;

- les abscisses de ces sommets appartiennent à la Farey d'ordre $\max(x_0, N - x_0)$;
- si le domaine a 4 sommets, deux d'entre eux ont la même abscisse ;
- les abscisses de deux sommets adjacents sont des fractions consécutives dans une série de Farey ;
- si l'abscisse d'un sommet est $\frac{p}{q}$, l'ordonnée de celui-ci est un multiple de $\frac{1}{q}$.

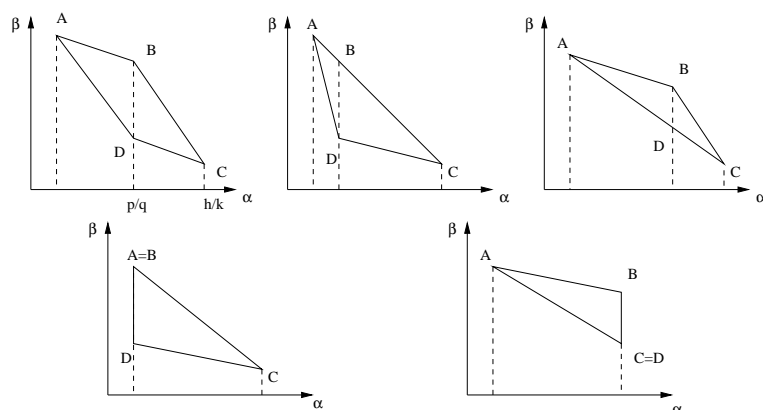


FIG. 2.15 — Différentes formes du domaine des droites dans le dual.

En considérant toutes ces propriétés, le domaine des solutions dans l'espace dual ne peut prendre que cinq formes différentes, représentées figure 2.15. Pour illustrer la quatrième propriété de ce domaine, dans le cas du quadrilatère de la figure 2.15, les fractions $\frac{p}{q}$ et $\frac{h}{k}$ sont adjacentes dans la série de Farey associée au domaine.

D'une manière encore plus précise, MCILROY (1985) introduit la notion *d'éventail de Farey* (de l'anglais *Farey fan*) permettant de décrire toutes les formes possibles de \bar{S} en fonction de l'ordre de Farey associé à la droite. En d'autres termes, un polygone de ce diagramme d'ordre q contenu dans le rectangle $[0, 1] \times [0, 1]$, appelé *cellule de Farey*, correspond à la pré-image d'un segment discret de longueur $q + 1$ (MCILROY, 1985; VITTONI, 1999). La figure 2.16 illustre ce calcul pour l'ordre 2, 3 et 6 : toutes les *cellules de Farey* contenues dans le rectangle $[0, 1] \times [0, 1]$ sont l'ensemble des domaines possibles pour tout ensemble de pixels générant un ordre de Farey de 2, 3 ou 6. Ainsi, dans la figure 2.16–(c), on remarque que tous ces polygones sont de la forme décrite dans le théorème 2.4.

Si nous oublions les propriétés arithmétiques du domaine dual, nous pouvons déjà proposer des algorithmes optimaux en temps et en mémoire pour la reconnaissance de segments discrets. En effet, étant donné que ce polygone des solutions ne peut avoir plus de 4 sommets, nous pouvons calculer la nouvelle forme de ce domaine, après insertion d'une nouvelle contrainte, en temps constant.

Cependant, la structure arithmétique nous permet de maintenir ce polygone sous forme irréductible, c'est-à-dire tel que tous les sommets s'écrivent sous forme de fractions irréductibles. Pour cela, MCILROY (1985) a décrit l'ensemble des coupures possibles après l'insertion d'un nouveau pixel :

Théorème 2.5 *Étant donné un ensemble de pixels \mathcal{S} connexes et la cellule de Farey qui lui est associé d'ordre q , si nous ajoutons un pixel qui engendre une contrainte dans l'espace des paramètres intersectant \bar{S} , celle-ci ne peut pas couper deux arêtes*

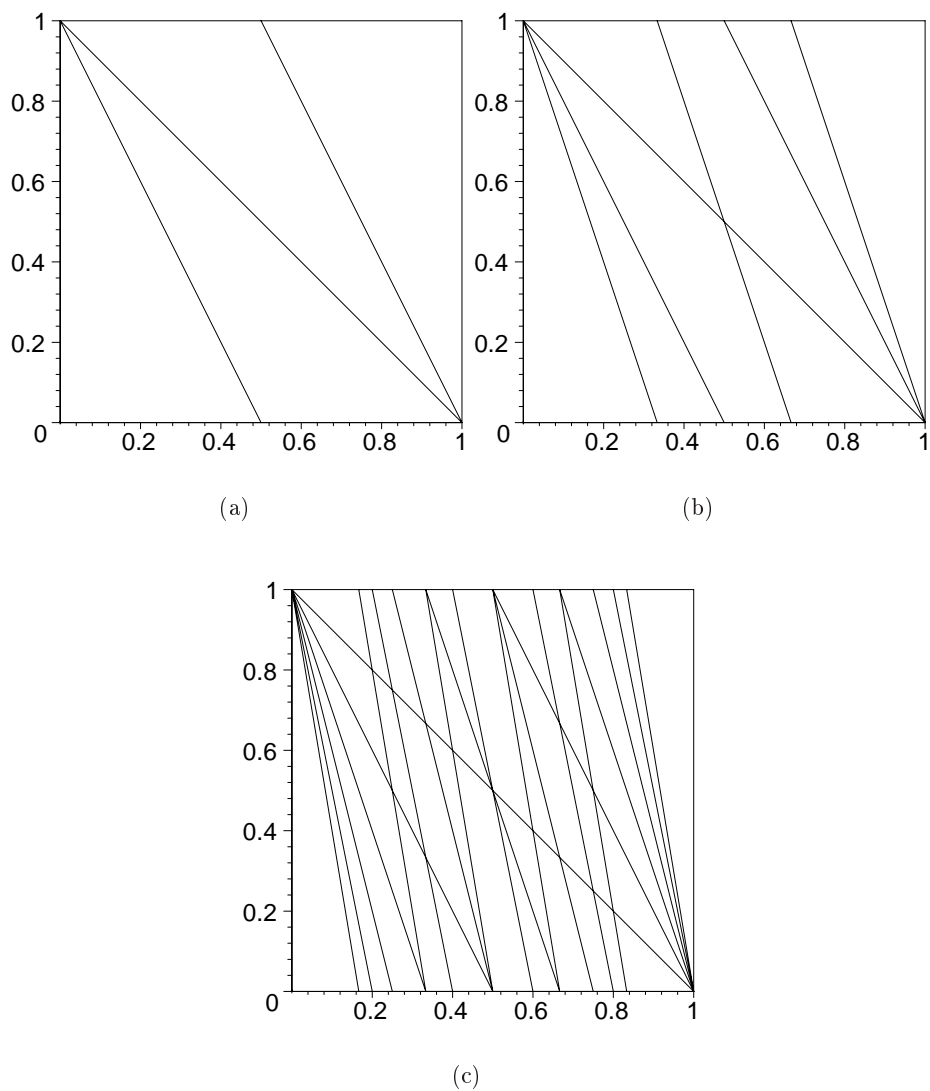


FIG. 2.16 –: *Illustration des éventails de Farey introduits par McILROY (1985) : (a) éventail d'ordre 2, (b) éventail d'ordre 3 et (c) éventail d'ordre 6. Les polygones définis dans le rectangle $[0, 1] \times [0, 1]$ correspondent aux domaines possibles associés aux segments discrets.*

adjacentes ni deux arêtes opposées. De plus si la contrainte coupe une arête, par exemple $[AB]$, du domaine, l'abscisse de cette intersection est le successeur de l'abscisse de A (resp. prédécesseur de l'abscisse de B) dans la série de Farey d'ordre $q + 1$.

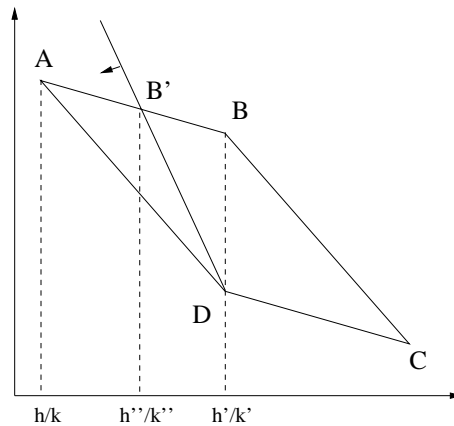


FIG. 2.17 –: L'ajout d'un nouveau point introduit une contrainte sur $\bar{\mathcal{S}}$ d'ordre q , le point B' a une abscisse qui est le médian des abscisses de A et B dans la série de Farey d'ordre $q + 1$.

Dans l'exemple illustré figure 2.17, la fraction $\frac{h''}{k''}$ est le successeur de $\frac{h}{k}$ et le prédécesseur de $\frac{h'}{k'}$ dans la série de Farey d'ordre $q + 1$. Étant donné que ce calcul de successeur se fait en temps constant, la mise à jour du domaine $\bar{\mathcal{S}}$ avec l'écriture sous forme irréductible des coordonnées des sommets se fait en temps constant.

En se basant sur ce principe, LINDENBAUM et BRUCKSTEIN (1993) proposent un algorithme optimal en temps, optimal en mémoire et incrémental avec un coût de mise à jour constant pour le problème de reconnaissance de droite discrète.

L'idée de cet algorithme est simple, lors de l'insertion des deux contraintes associées à un pixel, si le domaine est modifié :

1. Détection de l'arête coupée.
2. Calcul du point d'intersection en considérant le successeur de l'abscisse d'une des extrémités de l'arête dans la série de Farey d'ordre $q + 1$.
3. Déplacement des sommets pour maintenir un polygone $(ABCD)$.

Si nous insérons des pixels sans contrainte sur leur ordre ou leur connexité, la structure du domaine ne peut pas être contrôlée. Il peut être cependant intéressant d'avoir une écriture des coordonnées des sommets du polygone dans l'espace dual sous forme d'équations irréductibles. En effet, un point $(\frac{a}{b}, \frac{\mu}{b})$ solution du système permet la paramétrisation arithmétique du morceau de droite reconnue donnée par $\mathcal{D}(a, b, \mu)$ (dans le premier octant). Ainsi, lorsqu'une contrainte intersecte une arête, nous savons que si cette contrainte est issue d'un pixel connexe au morceau de droite déjà reconnu, l'ordre la série de Farey contenant la fraction n'augmente que de 1. Dans le cas général, l'augmentation de l'ordre de la série de Farey ne peut pas être bornée. Pour cela, VITTONI (1999); VITTONI et CHASSERY (2000) ont proposé l'utilisation d'un algorithme de GRABINER (1992) permettant de réduire la fraction en utilisant une dichotomie sur les fractions des extrémités de l'arête (voir algorithme 3). Même si la complexité au pire cas de cet algorithme est $O(\log(n))$ par réduction,

il est en pratique bien plus efficace qu'une réduction de fraction par l'algorithme d'Euclide. En effet, la clé de cet algorithme est de trouver l'ordre de Farey minimal qui contient la fraction en s'aidant d'un encadrement de celle-ci, ce qui est bien moins coûteux qu'Euclide en pratique si les pixels insérés ne sont pas *trop éloignés*.

Algorithme 3 Algorithme de GRABINER modifié

GRABINER_MODIFIÉ($I, \frac{p}{q}, \frac{r}{s}$)

{Calcul de l'intersection de l'inéquation I avec le segment $[\frac{p}{q}, \frac{r}{s}]$ }

- 1: $\frac{a}{b} = \frac{p}{q}$
 - 2: $\frac{c}{d} = \frac{r}{s}$
 - 3: **tant que** $I(\frac{a}{b}) + I(\frac{c}{d}) \neq 0$ **faire**
 - 4: **si** $I(\frac{a}{b}) + I(\frac{c}{d})$ et $I(\frac{p}{q})$ sont du même signe **alors**
 - 5: $\frac{a}{b} = \frac{a+c}{b+d}$
 - 6: $\frac{c}{d} = \frac{r}{s}$
 - 7: **sinon**
 - 8: $\frac{a}{b} = \frac{p}{q}$
 - 9: $\frac{c}{d} = \frac{a+c}{b+d}$
 - 10: **fin du si**
 - 11: **fin du tant que**
 - 12: **retourner** $\frac{a+c}{b+d}$
-

En se basant sur ce calcul de réduction de fraction, VITTONÉ (1999) présente un algorithme de reconnaissance dans le cas général mais avec une complexité au pire cas en $O(n^3 \log(n))$ très loin de la complexité effective de l'algorithme.

Cependant, nous pouvons adapter cet algorithme en utilisant, à la fois l'algorithme de GRABINER (1992) modifié par VITTONÉ (1999) et l'algorithme optimal de PREPARATA et SHAMOS (1985). Nous obtenons ainsi l'algorithme 4 dont la complexité est en $O(n \log^2(n))$. Cet algorithme est décrit en deux passes, la première pour construire le polytope des solutions et ensuite l'application de GRABINER pour réduire les sommets mais cette réduction peut bien évidemment s'insérer dans le calcul du polytope.

Algorithme 4 Ajout d'une contrainte avec réduction arithmétique, algorithme de VITTONÉ modifié

AJOUT_CONTRAINTE_VITTONÉ_MODIFIÉ(D, c)

{ D est le polygone déjà construit et c est la nouvelle contrainte}

- 1: Ajout de la contrainte avec l'algorithme de PREPARATA et SHAMOS (1985) (voir Annexe C)
 - 2: **pour tout** sommet p engendré par c **faire**
 - 3: Soit v et v' la séquence de sommets adjacents à p { v et v' sont sous forme irréductible car présents dans D }
 - 4: Réduction de la fraction de l'abscisse de p avec l'algorithme de GRABINER modifié sur les points v , p et v'
 - 5: **fin du pour**
-

2.2.2.2 Approche arithmétique

Nous présentons maintenant l'algorithme de reconnaissance proposé par DEBLED-RENNESON (1995); DEBLED-RENNESON et REVEILLÈS (1995) qui se base sur une

analyse des restes associés à une droite discrète arithmétique naïve $\mathcal{D}(a, b, \mu)$. Sans perte de généralité, nous nous plaçons dans le premier octant ($0 \leq a < b$).

Rappelons qu'un *point d'appui inférieur* (resp. *supérieur*) est un point vérifiant $ax - by = \mu + b - 1$ (resp. $ax - by = \mu$). Par ailleurs, un *point faiblement extérieur inférieur* (resp. *supérieur*) est un pixel qui vérifie $ax - by = \mu + b$ (resp. $ax - by = \mu + 1$).

Dans notre explication de l'algorithme de reconnaissance de droite discrète de DEBLED-RENNESON (1995), nous utiliserons les notations de VIALARD (1996a). En effet, ces notations sont équivalentes à celles de DEBLED-RENNESON et permettent une écriture plus simple dans le cas du calcul de tangente discrète que nous présenterons dans le paragraphe 5.3.

Ainsi, considérons un segment de droite discrète déjà reconnu Σ de paramètres a, b et μ . Nous noterons U et U' les points d'appui supérieurs du segment dont les abscisses sont respectivement minimale et maximale. De la même manière nous noterons L et L' les points d'appui inférieurs d'abscisses minimale et maximale (voir figure 2.18).

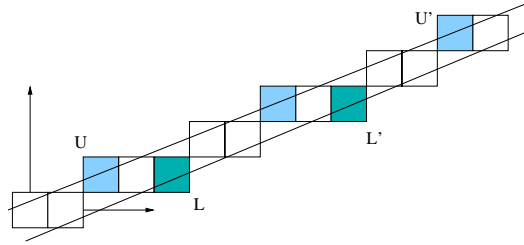


FIG. 2.18 —: Description des notations : U et U' sont les points d'appui supérieurs d'abscisses minimale et maximale, L et L' sont les points d'appui inférieurs d'abscisses minimale et maximale.

L'algorithme de reconnaissance de DEBLED-RENNESON considère l'ajout d'un point $M(x, y)$ de reste $r = ax - by$ au segment déjà reconnu. Cet algorithme repose sur le théorème suivant :

Théorème 2.6 (DEBLED-RENNESON (1995)) *Étant donné l'ajout d'un point M de reste $r = ax - by$ à droite d'un segment déjà reconnu Σ , et les points d'appui U, U', L et L' , nous avons :*

- si $\mu \leq r < \mu + b$ alors M appartient à la droite $\mathcal{D}(a, b, \mu)$ et donc $\Sigma \cup M$ est un segment discret
- si $r > \mu + b$ ou $r < \mu - 1$ alors $\Sigma \cup M$ n'est pas un segment de droite discrète
- si $r = \mu - 1$ alors M est faiblement extérieur à la droite discrète \mathcal{D} et donc $\overrightarrow{\Sigma \cup M}$ est un segment de droite discrète dont la pente est donnée par le vecteur \overrightarrow{UM}
- si $r = \mu + b$ alors M est faiblement extérieur à \mathcal{D} et donc $\Sigma \cup M$ est un segment de droite discrète dont la pente est donnée par \overrightarrow{LM}

Si l'ajout se fait à gauche du segment Σ , il suffit de considérer les vecteurs $\overrightarrow{U'M}$ et $\overrightarrow{L'M}$ au lieu de \overrightarrow{UM} et \overrightarrow{LM} .

La clef de ce théorème est la suivante : si nous ajoutons un point faiblement extérieur à un segment, nous devons modifier les paramètres de la droite reconnue tout en garantissant que tous les points de Σ déjà reconnus appartiennent aussi à cette nouvelle droite discrète.

D'une manière plus précise, si (b, a) désigne le vecteur directeur de la droite contenant Σ , l'ajout d'un point faiblement extérieur introduira le vecteur (b', a') . Or, M étant faiblement extérieur, les vecteurs (b, a) et (b', a') sont uni-modulaires, c'est-à-dire que nous avons :

$$ab' - ba' = 1$$

Pour conclure cette preuve, DEBLED-RENNESON utilise le théorème de MINKOWSKI qui stipule qu'il n'y a pas de points entiers dans le parallélogramme défini par des vecteurs uni-modulaires (voir (HARDY et WRIGHT, 1975)). Nous reprenons une illustration de VIALARD pour représenter les différents cas de ce théorème (voir figure 2.19).

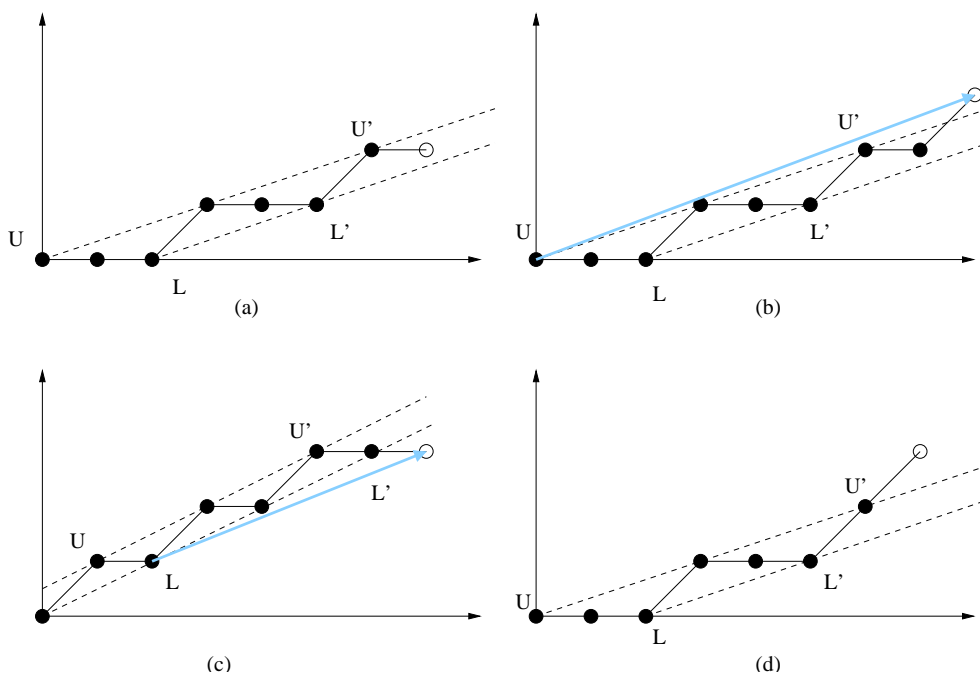


FIG. 2.19 – : Différents cas de figure lors de l'ajout d'un point à une droite discrète : (a) ajout d'un point ne changeant pas les paramètres, (b) ajout d'un point faiblement extérieur supérieur, dans ce cas le vecteur clair indique les nouveaux paramètres du segment discret, (c) ajout d'un point faiblement extérieur inférieur, le vecteur clair indique toujours le nouveau paramétrage de la courbe et (d) ajout d'un point fortement extérieur et mise en échec de la reconnaissance.

Nous pouvons maintenant décrire l'algorithme de reconnaissance de droite discrète de DEBLED-RENNESON (1995) (voir algorithme 5). Cet algorithme correspond à une modification apportée par VIALARD (1996a); BRAQUELAIRE et VIALARD (1999) à l'algorithme initial de DEBLED-RENNESON.

Cet algorithme est non seulement linéaire en le nombre de pixels de \mathcal{S} et optimal en mémoire, mais il offre aussi une possibilité d'implémentation très rapide.

En se basant sur une analyse identique, DEBLED-RENNESON (1995) propose aussi des algorithmes de reconnaissance de droites discrètes *épaisses* dans lesquelles les droites discrètes standards (4-connexes) sont un cas particulier. Dans ce cas

Algorithme 5 Algorithme de reconnaissance de droite discrète de DEBLED-RENNESON (1995)

RECONNAISSANCE_DEBLED(\mathcal{S}) {Algorithme modifié par VIALARD (1996a)}

```

1:  $M(x, y)$  et  $M'(x', y')$  sont les deux premiers points de  $\mathcal{S}$ 
2:  $a = y'$ ,  $b = 1$  et  $\mu = 0$ 
3:  $U = L = (0, 0)$ 
4:  $U' = L' = (1, y')$ 
5: SEGMENT=Vrai
6: tant que  $\mathcal{S}$  n'a pas été complètement parcouru et SEGMENT faire
7:    $M(x, y)$  = point suivant dans  $\mathcal{S}$ 
8:    $r = ax - by$ 
9:   si  $r < \mu - 1$  ou  $r > \mu + b$  alors
10:     SEGMENT=Faux
11:   sinon
12:     si  $r = \mu - 1$  ou  $r = \mu + b$  alors
13:       si  $r = \mu - 1$  alors
14:          $L = L'$ 
15:          $U' = M$ 
16:          $a = y - y_U$ 
17:          $b = x - x_U$ 
18:          $\mu = ax - by$ 
19:       fin du si
20:     si  $r = \mu + b$  alors
21:        $U = U'$ 
22:        $L' = M$ 
23:        $a = y - y_L$ 
24:        $b = x - x_L$ 
25:        $\mu = ax - by - b + 1$ 
26:     fin du si
27:   sinon
28:     si  $r = \mu$  alors
29:        $U' = M$ 
30:     sinon si  $r = \mu + b - 1$  alors
31:        $L' = M$ 
32:     fin du si
33:   fin du si
34: fin du tant que
35: fin du tant que

```

précis des droites standards, VIALARD (1996a) a illustré un passage direct entre la reconnaissance d'une droite 4-connexe et la reconnaissance d'une droite naïve (8-connexe) :

Proposition 2.4 (VIALARD (1996a)) *Soient a, b et μ dans \mathbb{Z} avec $0 \leq a < b$, le code de FREEMAN 4-connexe de la droite discrète standard $\mathcal{D}(a, b-a, \mu)$ est identique au code de FREEMAN 8-connexe de la droite discrète naïve $\mathcal{D}(a, b, \mu)$.*

Grâce à cette propriété, les algorithmes de reconnaissance de droites discrètes standards s'obtiennent à partir des algorithmes de reconnaissance des droites discrètes naïves par une simple réécriture dans le code de FREEMAN.

2.2.2.3 Comparaisons entre approche arithmétique et approche duale

Pour terminer cette analyse, nous montrons les liens entre la représentation d'une droite par son domaine des solutions dans le dual et la représentation sous forme arithmétique. Dans un premier temps, nous caractérisons dans l'espace primal, les différents sommets et arêtes du domaine associé à un segment discret dans le dual (voir DORST et SMEULDERS 1984; LINDENBAUM et BRUCKSTEIN 1993; VITTONI et CHASSERY 2000).

Nous considérons une cellule de Farey dont les sommets sont notés $(ABCD)$ comme indiqué dans la figure 2.17. Étant donné un segment de droite discrète de paramètre $\mathcal{D}(a, b, \mu)$, nous avons alors les propriétés (illustrées figure 2.20) :

Proposition 2.5

- La transformée de D dans l'espace primal est la droite d'appui supérieure associée au segment discret. D a pour coordonnées $(\frac{a}{b}, \frac{\mu}{b})$.
- La transformée de B dans l'espace primal est la droite d'appui inférieure, translaturée de 1 en y , associée au segment discret. B a pour coordonnées $(\frac{a}{b}, \frac{\mu+1}{b})$.
- La droite (AD) (resp. (DC)) correspond au point d'appui supérieur d'abscisse maximale U' (resp. minimale U).
- La droite (AB) (resp. (BC)) correspond au point d'appui inférieur, translaturé de 1 en y , d'abscisse minimale L (resp. maximale L').
- Le sommet A correspond dans le primal à la droite passant par U' et le translaturé de L de 1 en y .
- Le sommet C correspond dans le primal à la droite passant par U et le translaturé de L' de 1 en y .

Grâce à ces propriétés, nous pouvons construire entièrement la cellule de Farey associée à un segment discret à partir de sa caractérisation arithmétique. Réciproquement, à partir de la cellule de Farey nous pouvons obtenir les paramètres minimaux a et b tels que $\text{pgcd}(a, b) = 1$ et le paramètre de translation μ caractérisant les droites discrètes.

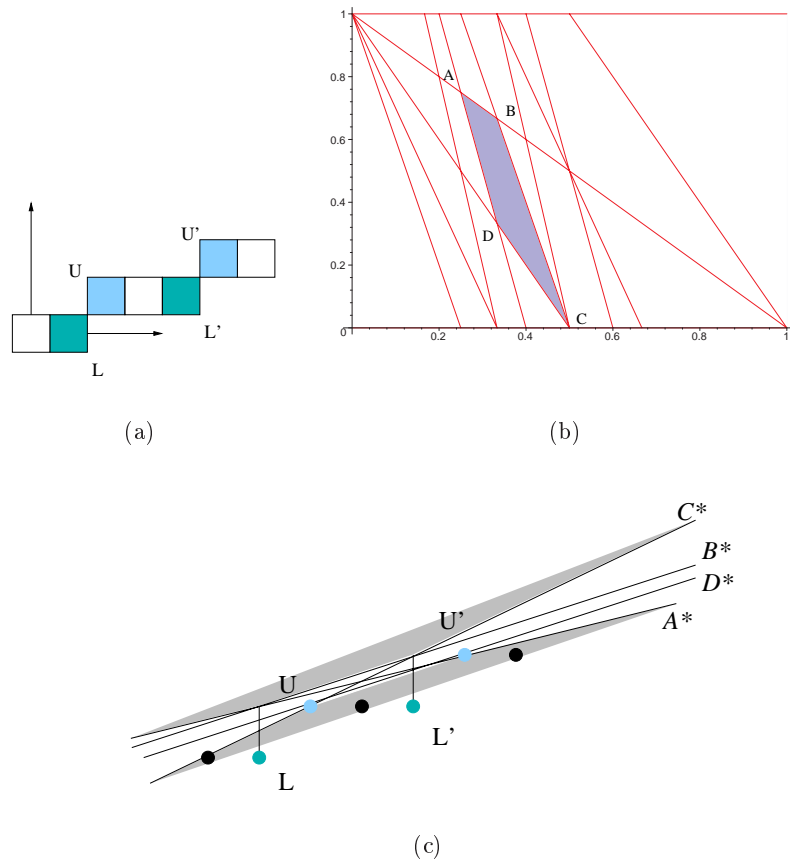


FIG. 2.20 – : Comparaisons entre approche arithmétique et approche basée sur l'espace des droites dans le dual : (a) segment de droite arithmétique $\mathcal{D}(1, 3, 1)$, (b) cellule de Farey associée au segment discret dans l'espace des paramètres et (c) représentation dans l'espace primal des éléments caractéristiques dans le dual, la partie claire correspond à l'ensemble des orientations des droites réelles se discrétisant dans le segment discret.

2.2.2.4 Bilan des algorithmes de reconnaissance présentés

Dans des utilisations pratiques de ces algorithmes de reconnaissance de droite discrète, le choix d'un algorithme peut ne pas être évident. Dans le tableau 2.2 nous résumons les différentes techniques présentées avec leur analyse en complexité et hypothèses.

2.2.2.5 De la reconnaissance à la segmentation de courbe

Comme nous l'avons illustré dans le tableau 2.1, le passage d'une droite discrète dans le premier octant vers les autres octants s'opère par symétries d'axes. Cependant, dans un contexte général, nous avons un ensemble de pixels \mathcal{E} sur lequel nous n'avons aucune information quant à l'octant auquel il appartient, le processus de reconnaissance ne peut donc s'appliquer directement.

Nous considérons un ensemble de pixels \mathcal{E} et nous voulons savoir quel est le plus long segment de droite discrète qu'il est possible de reconnaître à partir d'un pixel M de \mathcal{E} , dans une direction donnée. L'idée pour résoudre ce problème se décompose en deux étapes :

1. à partir de M , détecter dans quel octant se trouve le morceau de courbe de \mathcal{E} ;
2. une fois l'octant connu, on utilise le processus de reconnaissance de droite discrète dans le premier octant et le tableau des symétries 2.1.

A partir d'un point, calculer dans quel octant se trouve un morceau de courbe de \mathcal{E} localement en M se fait simplement en regardant le code de FREEMAN. Dès que nous avons repéré dans ce codage deux codes distincts consécutifs, nous savons dans quel octant se trouve la courbe en partant de M . Même si ce calcul est trivial sur le plan théorique, l'implémentation en temps optimal de ce processus n'est pas évident. Pour cela, DEBLED-RENNESON et REVEILLÈS (1995) ont proposé un algorithme pour le problème plus général de la segmentation de courbe en droites discrètes arithmétiques en temps optimal en implémentant ce processus de découverte d'octant en même temps que celui de la reconnaissance.

Le processus de segmentation considère une courbe discrète \mathcal{E} , un point de départ et un sens de parcours. Elle construit la segmentation de cette courbe en morceaux de droites discrètes de longueurs maximales. Ainsi, en partant d'un point M , on va reconnaître le plus grand morceau de droite discrète de \mathcal{E} . Une fois la reconnaissance terminée, on initialise un nouveau processus de reconnaissance à partir du premier point non reconnu (voir figure 2.21). Cette segmentation permet une description polygonale de la courbe discrète tout en conservant des propriétés de réversibilité : on peut trivialement reconstruire l'ensemble \mathcal{E} à partir de sa segmentation en segments discrets.

Nous verrons au chapitre 5 l'utilisation de cet outil pour l'estimation de la longueur d'une courbe discrète.

2.3 Droites discrètes 3D

Nous regardons maintenant la construction et les propriétés des droites discrètes tri-dimensionnelles. La synthèse et la reconnaissance de ces objets sont utilisées dans de très nombreuses applications. Par exemple, dans le contexte de la synthèse d'image

Algorithme	Complexité en temps	Incémental	Mémoire	Hypothèses	Remarques
MEGIDDO (1984)	$O(n)$	oui, voir BUZER (2002)	-	Aucune	linéarité uniquement
PREPARATA et SHAMOS (1985)	$O(n \log(n))$	oui	$O(n^2)$	Aucune	
VITTONÉ (1999)	$O(n^3 \log(n))$	oui	$O(n^2)$	Aucune	coordonnées sous forme irréductible
VITTONÉ modifié (alg. 4)	$O(n \log^2(n))$	oui	$O(n^2)$	Aucune	coordonnées sous forme irréductible
O'ROURKE (1981)	$O(n)$	oui	$O(n)$	pixels triés selon un axe	
DORST et SMEULDERS (1991a); LINDENBAUM et BRUCKSTEIN (1993)	$O(n)$	oui en $O(1)$ par ajout	$O(1)$	connexité	coordonnées sous forme irréductible
DEBLED-RENNESON (1995)	$O(n)$	oui en $O(1)$ par ajout	$O(1)$	connexité	caractérisation arithmétique

TAB. 2.2 – : Algorithmes et complexités pour la reconnaissance de droites discrètes.

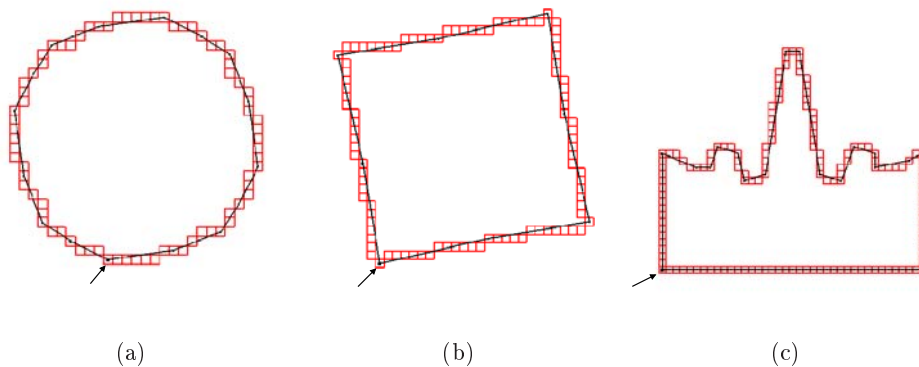


FIG. 2.21 – : *Exemples de segmentation d'une courbe discrète en droites discrètes avec l'algorithme de DEBLED-RENNESON et REVEILLÈS (1995) : la courbe polygonale noire correspond au résultat de la segmentation et la flèche indique le point départ, (a) un cercle discret, (b) un carré avec une rotation de 10 degrés et (c) sur une courbe construite par une fonction sinus cardinale.*

par lancé de rayons, la construction de rayons discrets 3D a permis l'accélération de nombreuses techniques.

Nous commençons par présenter les différentes approches existantes pour le tracé de droites discrètes 3D. Puis nous présentons les propriétés de ces droites ainsi que les algorithmes permettant leur reconnaissance.

2.3.1 Caractérisation et propriétés des droites discrètes 3D

Comme nous l'avons présenté précédemment, le tracé de droites dans des grilles 3D a été très développé dans le cadre de la synthèse d'images par lancé de rayons (AMANATIDES et WOO, 1987; KAUFMAN, 1987; VIDAL, 1992; SALAM ET AL., 1999). Le problème s'écrit en ces termes : étant donné deux points d'une grille dans \mathbb{Z}^3 , comment tracer la droite joignant ces deux points.

Pour résoudre ce problème, deux approches sont possibles.

Le tracé par projection consiste en une décomposition du problème en dimension 2. Ainsi, le principe consiste à tracer deux droites discrètes 2D dans deux plans canoniques de la grille pour ensuite construire la droite 3D par projection inverse (AMANATIDES et WOO, 1987).

Le tracé direct consiste en un tracé de la droite en 3D directement. Ces approches se basent sur une généralisation 3D des algorithmes 2D classiques comme celui de BRESENHAM (KAUFMAN et SHIMONY, 1986; COHEN et KAUFMAN, 1991).

Sur le plan de la caractérisation de tels objets, de nombreux auteurs ont proposé des définitions de droites discrètes 26-connexes (KIM, 1983; STOJMENOVIC et TOSIC, 1991). Dans cette partie, nous présentons la caractérisation arithmétique de ces droites proposées par DEBLED-RENNESON (1995) et reprise par COEUR-JOLLY ET AL. (2001a) dans le cadre de l'estimation de longueur d'une courbe 3D par segmentation en droite discrète.

Nous considérons la droite discrète 3D de vecteur directeur $(a, b, c)^T \in \mathbb{Z}^3$ appartenant dans le premier 48^{ème} d'espace, c'est-à-dire tel que $a \geq b \geq c > 0$. Nous

discuterons dans la section suivante la généralisation à tout l'espace.

Définition 2.5 (Droite discrète 3D (DEBLED-RENNESON, 1995)) *La droite discrète 3D, notée $\mathcal{D}_{3D}(a, b, c, \mu, \mu', e, e')$, dont le vecteur directeur $V(a, b, c)$ est dans le premier 48^{ème} d'espace, est définie comme étant l'ensemble des voxels (x, y, z) de \mathbb{Z}^3 vérifiant les inéquations diophantiennes suivantes :*

$$\mathcal{D} \begin{cases} \mu \leq cx - az < \mu + e \\ \mu' \leq bx - ay < \mu' + e' \end{cases}$$

avec μ, μ', e et e' dans \mathbb{Z} . μ et μ' sont les **bornes inférieures** de \mathcal{D}_{3D} , e et e' représentent l'**épaisseur arithmétique** de \mathcal{D}_{3D} .

Une définition arithmétique plus générale a été proposée par FIGUEIREDO et REVEILLÈS (1995).

Comme nous pouvons le remarquer, la projection la droite \mathcal{D}_{3D} sur le plan Oxz correspond à la droite discrète 2D $\mathcal{D}(c, a, \mu, e)$. De la même manière, la projection de cette droite 3D sur le plan Oxy correspond à la droite 2D $\mathcal{D}(b, a, \mu', e')$.

Nous définissons les droites discrètes naïves 3D de la façon suivante :

Définition 2.6 (Droite discrète naïve 3D (DEBLED-RENNESON, 1995)) *une droite discrète 3D de paramètres $\mathcal{D}_{3D}(a, b, c, \mu, \mu', e, e')$ dans le premier 48^{ème} d'espace est dite **naïve** si et seulement si l'épaisseur vérifie $e = e' = a$.*

D'après la remarque précédente, la droite 3D se projette bijectivement sur les plans Oxy et Oxz en deux droites naïves 2D. Par conséquence, de la même manière qu'en 2D, nous avons un théorème de structure des droites discrètes 3D en fonction de l'épaisseur arithmétique :

Théorème 2.7 (connexité des droites discrètes 3D (DEBLED-RENNESON, 1995)) *Considérons une droite discrète $\mathcal{D}_{3D}(a, b, c, \mu, \mu', e, e')$ avec $a > b > c > 0$, alors*

- si $e \geq a + c$ et $e' \geq a + b$, \mathcal{D}_{3D} est 6-connexe (voir figure 2.22).
- si $e \geq a + c$ et $a \leq e' < a + b$, ou $e' \geq a + b$ et $a \leq e < a + c$, \mathcal{D}_{3D} est 18-connexe
- si $a \leq e < a + c$ et $a \leq e' < a + b$, \mathcal{D}_{3D} est 26-connexe (voir figure 2.23.)
- si $e < a$ ou $e' < a$, \mathcal{D}_{3D} est déconnectée.

La preuve de ce théorème se base sur une analyse des projections de la droite 3D et sur le théorème 2.1 de connexité des droites discrètes 2D (DEBLED-RENNESON, 1995).

2.3.2 Tracé de droites discrètes 3D

Nous détaillons ici les différents algorithmes de tracé de droite discrète naïve 3D. Nous présentons tout d'abord l'algorithme de COHEN et KAUFMAN (1991) qui correspond à une généralisation tridimensionnelle de l'algorithme de BRESENHAM (1965) (voir algorithme 6). Ceux-ci considèrent deux voxels de la grille et tracent le segment de droite 26-connexe les joignant.

En se basant sur la caractérisation arithmétique, nous présentons l'algorithme de tracé proposé par DEBLED-RENNESON (1995) (voir algorithme 7). Ce tracé

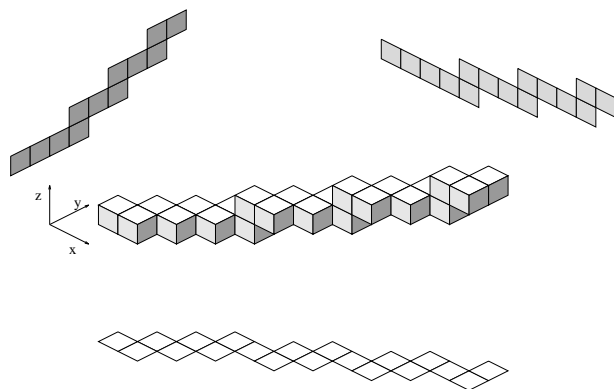


FIG. 2.22 –: Droites discrètes 3D de paramètres $\mathcal{D}_{3D}(10, 7, 3, 0, -9, 13, 17)$, conformément au théorème 2.7, cette droite est 6-connexe. De plus, les droites 2D dans les plans Oxy et Oxz sont 4-connexes (DEBLED-RENNESON, 1995).

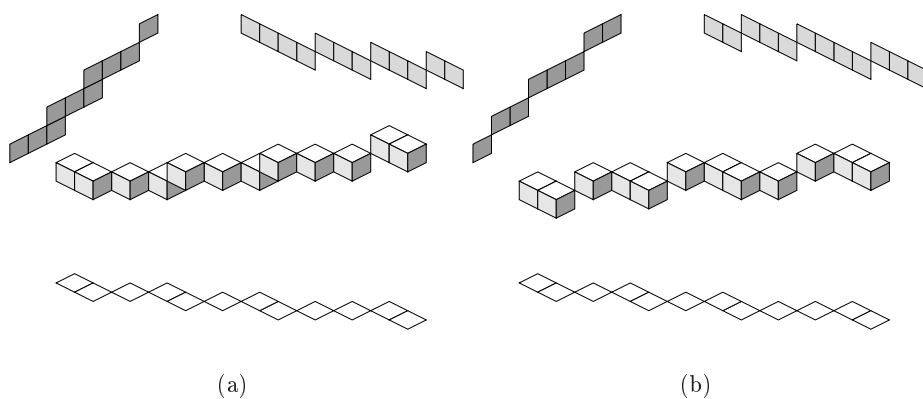


FIG. 2.23 –: Droites discrètes 3D 26-connexes : (a) droite naïve $\mathcal{D}_{3D}(10, 7, 3, 0, 0)$ et (b) droite naïve $\mathcal{D}_{3D}(10, 7, 3, -5, 0)$ (DEBLED-RENNESON, 1995).

Algorithme 6 Tracé de droites discrètes 3D de COHEN et KAUFMAN (1991)

```

TRACÉ_3D_COHEN_KAUFMAN(A,B)
  {avec  $A, B \in \mathbb{Z}^3$  tels que  $x_B - x_A > 0, y_B - y_A > 0$  et  $z_B - z_A > 0$ }
1:  $v_x = x_B - x_A, v_y = y_B - y_A, v_z = z_B - z_A;$ 
2:  $s_x = \text{sgn}(v_x), s_y = \text{sgn}(v_y), s_z = \text{sgn}(v_z)$ 
3:  $a_x = \text{abs}(v_x), a_y = \text{abs}(v_y), a_z = \text{abs}(v_z)$ 
4:  $b_x = 2a_x, b_y = 2a_y, b_z = 2a_z$ 
5:  $e_{xy} = a_y - a_x, e_{xz} = a_z - a_x, e_{zy} = a_y - a_z$ 
6:  $x = x_A, y = y_A, z = z_A$ 
7:  $n = a_x + a_y + a_z$ 
8: tant que  $n \neq 0$  faire
9:    $n- = 1$ 
10:  Affiche_point( $x, y, z$ )
11:  si  $e_{xy} < 0$  alors
12:    si  $e_{xz} < 0$  alors
13:       $x+ = s_x$ 
14:       $e_{xy}+ = b_y, e_{xz}+ = b_z$ 
15:    sinon
16:       $z+ = s_z$ 
17:       $e_{zy}+ = b_y, e_{xz}- = b_x$ 
18:    sinon si  $e_{zy} < 0$  alors
19:       $z+ = s_z$ 
20:       $e_{zy}+ = b_y, e_{xz}- = b_x$ 
21:    sinon
22:       $y+ = s_y$ 
23:       $e_{xy}- = b_x, e_{zy}- = b_z$ 
24:    fin du si
25:  fin du si
26: fin du tant que

```

Algorithme 7 Tracé de droites discrètes 3D de DEBLED-RENNESON (1995)

```

TRACÉ_3D_DEBLED(A,B, $\mu, \mu'$ )
  {avec  $A, B \in \mathbb{Z}^3$  tels que  $x_B - x_A > y_B - y_A > 0$  et  $y_B - y_A > z_B - z_A > 0$ }
1:  $v_x = x_B - x_A, v_y = y_B - y_A, v_z = z_B - z_A$ 
2:  $r_1 = v_z * x_A - v_x * z_A - \mu$ 
3:  $r_2 = v_y * x_A - v_x * y_A - \mu'$ 
4:  $x = x_A, y = y_A, z = z_A$ 
5: tant que  $x < x_B$  faire
6:    $x = x + 1$ 
7:    $r_1 = r_1 + v_z$ 
8:    $r_2 = r_2 + v_y$ 
9:   si  $r_1 < 0$  ou  $r_1 \geq v_x$  alors
10:      $z = z + 1$ 
11:      $r_1 = r_1 - v_x$ 
12:   fin du si
13:   si  $r_2 < 0$  ou  $r_2 \geq v_x$  alors
14:      $y = y + 1;$ 
15:      $r_2 = r_2 - v_x$ 
16:   fin du si
17:  Affiche_point( $x, y, z$ )
18: fin du tant que

```

considère deux paramètres supplémentaires que sont les bornes inférieures μ et μ' . Ainsi, si nous considérons $\mu = \mu' = -\left\lceil \frac{\max(|vx|, |vy|, |vz|)}{2} \right\rceil + 1$, nous nous ramenons au tracé de BRESENHAM (1965) étendu en 3D par COHEN et KAUFMAN (1991).

Dans la majorité des cas, ces tracés de droites s'effectuent pour un vecteur directeur $(a, b, c)^T$ tel que $0 \leq a < b < c$. En d'autres termes, la synthèse de la droite s'effectue dans le premier $48^{\text{ème}}$ d'espace et pour avoir une droite dans une autre orientation, il faut appliquer des symétries d'axes. Le problème majeur est donc l'implémentation efficace de ces algorithmes sachant qu'il est inconcevable de coder toutes les transformations en fonction du vecteur directeur comme cela était le cas en 2D avec le tableau 2.1 (voir figure 2.24).

Nous utilisons donc une technique présentée par REVEILLÈS (2001) pour calculer la matrice de rotation qui permet d'envoyer tout vecteur dans un d'un certain $48^{\text{ème}}$ d'espace vers le premier (*e.g.* tel que $0 \leq a < b < c$). Plus formellement, REVEILLÈS (2001) propose une implémentation de ce groupe des symétries du cube, noté O_h . L'idée est donc de calculer la transformation $g \in O_h$ telle que, pour tout vecteur v , $g.v$ soit dans le premier $48^{\text{ème}}$ d'espace (domaine fondamental noté F , voir figure 2.24). De manière évidente, nous avons $g^{-1}.g.v = v$ ce qui nous permet de retrouver le vecteur v après transformation inverse.

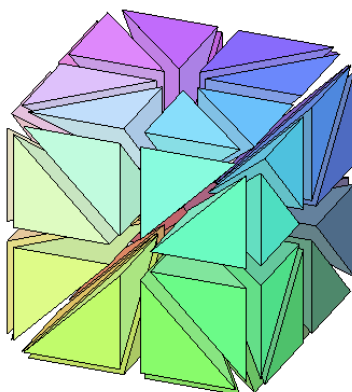


FIG. 2.24 –: Décomposition de l'espace en $48^{\text{ème}}$, représentation graphique du groupe O_h (REVEILLÈS, 2001).

L'algorithme de REVEILLÈS propose une implémentation de ce groupe O_h en utilisant le groupe \mathcal{E}_3 des permutations de trois lettres. L'algorithme 8 décrit cette méthode très simple à implémenter.

Pour illustrer cet algorithme, nous considérons le vecteur $v = (-3, 17, 5)$. Nous souhaitons trouver la transformation permettant d'envoyer ce vecteur vers F . À la suite de la première étape l'algorithme, nous avons la matrice :

$$M = \begin{bmatrix} -3 & 17 & 5 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Algorithme 8 Implémentation du groupe O_h pour le changement de 48^{ème} d'espace de REVEILLÈS (2001)

IMPLEMENTATION_ O_h _REVEILLÈS(a, b, c) $\{a, b$ et c composent le vecteur $v\}$

1: Construction d'une matrice 4×3 notée M telle que :

- la première ligne est (a, b, c)
- le bloc 3×3 en dessous est la matrice unité $I_{3 \times 3}$

2: Multiplication de chaque colonne de M par le signe des éléments de la première ligne

3: Permutation des colonnes pour que les éléments de la première ligne soient triés

4: Le bloc 3×3 sous la première ligne est solution

A la seconde étape, nous faisons *tomber* les signes dans les colonnes :

$$M = \begin{bmatrix} 3 & 17 & 5 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Nous trions les colonnes par rapport à la première ligne et obtenons la matrice :

$$M = \begin{bmatrix} 3 & 5 & 17 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Finalement, nous avons :

$$g = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Nous pouvons vérifier : $g \cdot (-3, 17, 5) = (3, 5, 17)$. De plus $g^{-1} = g^T$ et $g^{-1}(3, 5, 17) = (-3, 17, 5)$.

Cette technique très simple nous permet donc de généraliser le tracé de droite des algorithmes 6 et 7 à tout l'espace.

2.3.3 Reconnaissance et segmentation

Nous décrivons maintenant les algorithmes de reconnaissance de droites discrètes 3D. L'algorithme de reconnaissance que nous présentons est celui de DEBLED-RENNESON (1995) basé sur l'écriture arithmétique des droites discrètes 3D. Cet algorithme se base sur les propriétés de projection sur les plans d'axe des droites discrètes naïves 3D. Ainsi, étant donné un ensemble de voxels 26-connexes, noté \mathcal{E} , l'algorithme 9 permet de décider si \mathcal{E} est un morceau de droite discrète 3D. De plus, cet algorithme nous donne la paramétrisation arithmétique de \mathcal{E} si celui-ci est un segment discret.

Bien évidemment, n'importe quel algorithme de reconnaissance de droite discrète peut être utilisé à l'étape 5 de cet algorithme, sous réserve qu'il soit possible de

1. A^T représente la transposée de la matrice A .

Algorithme 9 Reconnaissance de segment de droite discrète naïve 3D de DEBLED-RENNESON (1995)

```

RECONNAISSANCE_DROITE_3D( $\mathcal{E}$ )    { $\mathcal{E}$  est un ensemble de voxels 26-connexes}
1: si la projection de  $\mathcal{E}$  ne se projette pas bijectivement sur au moins deux des trois
   plans d'axe notés  $P_1$  et  $P_2$  alors
2:   retourner Faux    { $\mathcal{E}$  n'est pas une droite discrète naïve 3D}
3: sinon
4:   soient  $C_1$  et  $C_2$  les courbes projetées de  $\mathcal{E}$  dans les plans  $P_1$  et  $P_2$ 
5:   si  $C_1$  et  $C_2$  sont des droites discrètes naïves 2D alors
6:     soient  $\mathcal{D}(c_1, a_1, \mu_1)$  les paramètres de  $C_1$ 
7:     soient  $\mathcal{D}(b_2, a_2, \mu_2)$  les paramètres de  $C_2$ 
8:     soit  $m = \text{ppcm}(a_1, a_2)$ 
9:     soient  $k_1$  tel que  $m = k_1 a_1$  et  $k_2$  tel que  $m = k_2 a_2$ 
10:    retourner  $\mathcal{D}_{3D}(m, k_2 b_2, k_1 b_1, \mu_1, \mu_2)$ 
11:   sinon
12:    retourner Faux    { $\mathcal{E}$  n'est pas une droite discrète naïve 3D}
13:   fin du si
14: fin du si

```

recalculer les paramètres arithmétiques des droites. Cependant, pour garder une forme unifiée nous utilisons l'algorithme 5 de DEBLED-RENNESON (1995). La complexité de cet algorithme est optimale en temps, c'est-à-dire $O(n)$ si \mathcal{E} contient n voxels et $O(1)$ en mémoire.

Nous souhaitons maintenant construire une segmentation d'une courbe discrète 3D en morceaux de droites discrètes. Pour cela, il nous faut détecter localement les plans de projection. Nous considérons donc le processus suivant : étant donné un segment 3D déjà reconnu, nous ajoutons un voxel M en maintenant valides deux contraintes, la première est que ce point M se projette bijectivement sur au moins deux plans d'axe et la seconde est que sur ces plans valides, le projeté de M appartient au segment naïf 2D de la projection du segment 3D déjà reconnu. Nous obtenons ainsi l'algorithme 10 qui nécessite un algorithme de reconnaissance de droite 2D capable de gérer des changements d'octants. Cet algorithme est aussi optimal en temps et en mémoire.

Algorithme 10 Segmentation en segments de droites discrètes naïves 3D de DEBLED-RENNESON (1995)

```

SEGMENTATION_DROITE_3D( $\mathcal{E}$ )    { $\mathcal{E}$  est un ensemble de voxels 26-connexes}
1: soit  $M$  le premier point de  $\mathcal{E}$ 
2:  $M$  est un sommet de la segmentation
3: tant que  $\mathcal{E}$  n'a pas été entièrement parcouru faire
4:   Initialiser la reconnaissance 2D sur les plans d'axe
5:   tant que  $M$  se projette bijectivement dans deux des trois plans et que les
   projections sont des droites naïves 2D dans ces plans faire
6:     soit  $M$  le voxel suivant dans  $\mathcal{E}$ 
7:   fin du tant que
8:    $M$  est un sommet de la segmentation
9: fin du tant que

```

La figure 2.25 présente des résultats de segmentation de courbes discrètes 26-connexes.

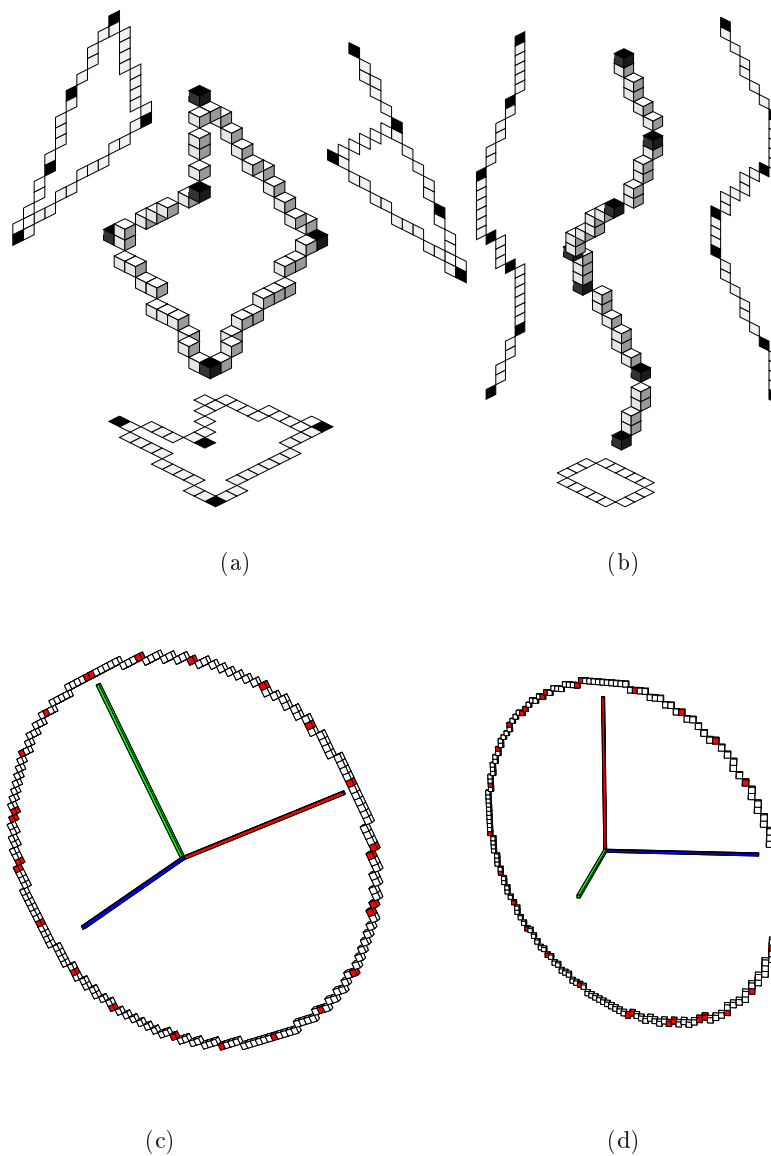


FIG. 2.25 – Exemples de segmentation de courbes 26-connexes en segments de droite naïve 3D : les voxels sombres correspondent aux sommets de la segmentation, (a) et (b) courbes 26-connexes synthétiques, (c) segmentation d'un cercle 3D et (d) segmentation d'une ellipse 3D (DEBLED-RENNESON, 1995; COEURJOLLY ET AL., 2001a)

2.4 Plans Discrets

Nous nous intéressons maintenant à la notion de plan discret et aux différents algorithmes permettant de les reconnaître. Dans les paragraphes précédents, nous avons vu que, dans le cas des droites discrètes, nous avons une caractérisation arithmétique très forte, que ce soit dans l'espace dual ou dans l'espace primal. De plus, des algorithmes très efficaces existent pour la reconnaissance ou la segmentation d'une courbe en droites discrètes.

Dans le cas tridimensionnel, le problème est plus complexe. L'analyse et la reconnaissance de ces objets sont des domaines de recherche bien plus récents que pour les droites et de nombreux problèmes restent ouverts.

Dans une première partie, nous commençons par décrire les différentes définitions des plans discrets proposées dans la littérature. Ensuite nous présentons les solutions algorithmiques associées pour la reconnaissance de plans discrets. Puis nous nous intéressons au processus de facettisation d'un objet discret.

2.4.1 Définitions et propriétés

De la même manière qu'en dimension 2, il existe de nombreuses approches pour définir un plan discret. Une fois encore, ces définitions sont généralement liées à un processus de discrétisation du plan euclidien. En considérant par exemple une discrétisation GIQ, on appellera *plan discret* l'ensemble des voxels (x, y, z) vérifiant :

$$z = [\alpha x + \beta y + \gamma] \quad (2.3)$$

pour (α, β, γ) dans \mathbb{R}^3 .

En se basant sur ce processus de discrétisation, KIM (1984b) propose une caractérisation des plans discrets à l'aide d'une propriété similaire à la propriété de corde de ROSENFELD (définition 2.1). Il utilise la notion de *triangle corde*, illustrée figure 2.26 (KIM et ROSENFELD, 1982).

Définition 2.7 (Propriété du triangle corde (KIM et ROSENFELD, 1982)) *Un ensemble de voxels \mathcal{E} vérifie la propriété du triangle corde si pour tout triangle T formé de u, v et w dans \mathcal{E} et pour tout point (x, y, z) de T , il existe un point p de \mathcal{E} tel que :*

$$\max\{|p_x - x|, |p_y - y|, |p_z - z|\} < 1$$

Grâce à cette définition, KIM (1984b) montre que cette propriété permet de caractériser un plan discret si celui-ci est infini et sans bord. Cependant, dans le cas d'un morceau de plan discret, cette propriété n'est ni une condition suffisante, ni une condition nécessaire.

Il montre le théorème suivant faisant le lien entre les morceaux de plans discrets et des propriétés sur leur enveloppe convexe (voir figure 2.27):

Théorème 2.8 (Plan discret et enveloppe convexe KIM (1984b)) *Un ensemble de voxels \mathcal{E} est un morceau de plan discret si et seulement si il existe un plan \mathcal{P} s'appuyant sur une face de l'enveloppe convexe de \mathcal{E} tel que :*

$$d_H(\mathcal{E}, \mathcal{P}) < 1$$

avec d_H la distance de HAUSDORFF adaptée à la grille :

$$d_H(\mathcal{E}, \mathcal{P}) = \min\{\max_{v \in \mathcal{E}}(d_x(v, \mathcal{P})), \max_{v \in \mathcal{E}}(d_y(v, \mathcal{P})), \max_{v \in \mathcal{E}}(d_z(v, \mathcal{P}))\}$$

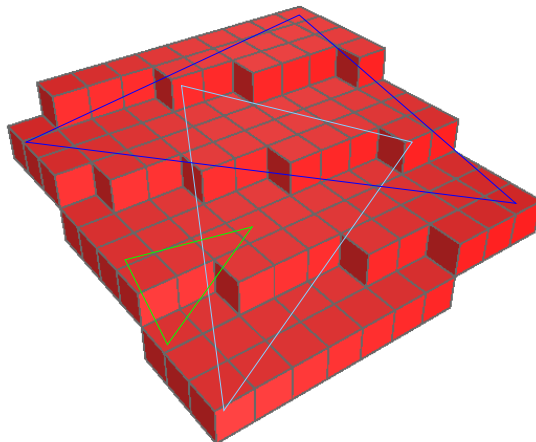


FIG. 2.26 –: Illustration de la propriété du triangle corde avec un morceau de plan discret et trois triangles vérifiant cette propriété.

où d_x correspond à la distance en x entre un voxel $v(x, y, z)$ et le plan \mathcal{P} , donnée par $d_x = |x - x'|$ pour (x', y, z) sur \mathcal{P} . d_y et d_z sont donnés de la même manière.

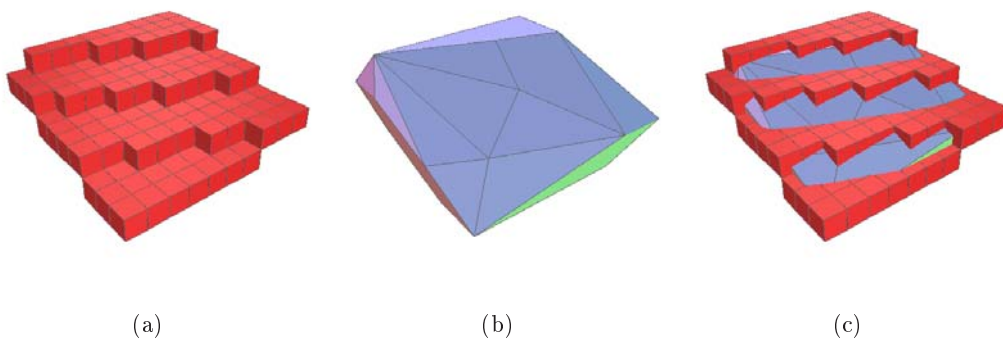


FIG. 2.27 –: Illustration du théorème de KIM (1984b) : (a) plan discret donné par l'équation 2.3, (b) enveloppe convexe des voxels et (c) les deux volumes dans le même repère.

DEBLED-RENNESON (1995) a cependant montré un cas négligé par ce théorème. Ce contre-exemple correspond à la discrétisation du plan euclidien $5x + 9y - 29z = 0$ (voir figure 2.28). Dans ce cas, aucun plan s'appuyant sur une facette de l'enveloppe convexe de cet ensemble de voxels n'est à une distance inférieure à 1 de tous les voxels.

DEBLED-RENNESON (1995) propose donc une reformulation de ce théorème en ajoutant une condition pour qu'un ensemble de voxels soit un morceau de plan :

Un ensemble de voxels \mathcal{E} est un morceau de plan discret si et seulement si le plan

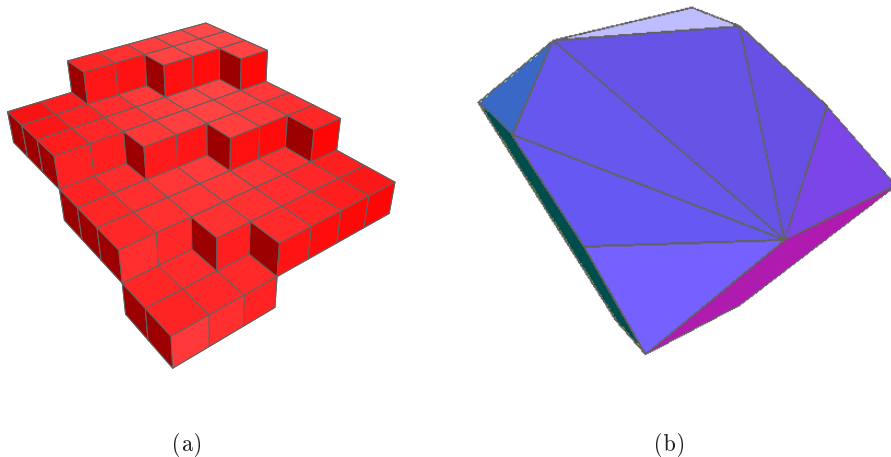


FIG. 2.28 –: Cas négligé dans le théorème de KIM (1984b) présenté par DEBLED-RENNESON (1995) : (a) morceau de plan qui correspond à la discrétisation de $5x + 9y - 29z = 0$ pour $0 \leq x \leq 6$ et $0 \leq y \leq 7$ et (b) enveloppe convexe des points.

\mathcal{P} défini par deux arêtes de l'enveloppe convexe de \mathcal{E} vérifie :

$$d_H(\mathcal{E}, \mathcal{P}) < 1$$

Si les deux arêtes en question appartiennent à la même face, nous nous ramenons au théorème initial de KIM.

Une autre caractérisation des plans discrets a été proposée par VEELAERT (1993, 1994). Celle-ci se base sur une généralisation de la propriété de régularité de HUNG (1985) en 2D.

Définition 2.8 (Propriété de régularité des plans discrets VEELAERT (1993))

Un ensemble voxels \mathcal{E} est dit régulier (ou even) si et seulement si :

- la projection de \mathcal{E} sur le plan Oxy est bijective
- pour tout quadruplet (u, v, w, t) de voxels de \mathcal{E} tel que $(v\vec{u}_x, v\vec{u}_y) = (t\vec{w}_x, t\vec{w}_y)$, nous avons :

$$|v\vec{u}_z - t\vec{w}_z| < 1$$

Ensuite, VEELAERT (1993, 1994) montre que cette propriété de régularité est nécessaire et suffisante pour caractériser tout morceau de plan discret dont la projection sur l'axe Oxy est un rectangle. Il montre aussi cette équivalence pour un ensemble de voxels dont la projection sur Oxy est un rectangle *partiellement étendu* ce qui correspond intuitivement à un rectangle auquel on ajouterait une ligne ou une colonne non complète (voir figure 2.29).

Pour terminer cette étude sur les différentes caractérisations des plans, nous présentons l'approche arithmétique. La définition d'un plan discret arithmétique a été proposée par ANDRÈS (1993) et se base sur une généralisation en 3D de la notion de droite discrète arithmétique :

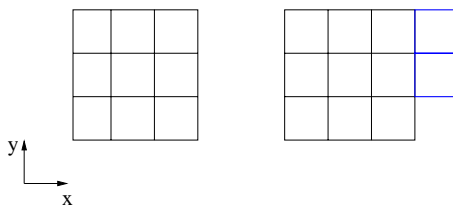


FIG. 2.29 –: *Rectangle et rectangle partiellement étendu en x de VEELAERT (1994).*

Définition 2.9 (Plan discret arithmétique (ANDRÈS, 1993)) *Un ensemble de voxels \mathcal{E} appartient à un plan discret arithmétique de vecteur normal $(a, b, c)^T$, de borne inférieure μ et d'épaisseur ω (avec a, b, μ et ω dans \mathbb{Z} , et $\text{pgcd}(a, b, c) = 1$), si et seulement si tous les voxels (x, y, z) de \mathcal{E} vérifient la double inéquation diophantienne :*

$$\mu \leq ax + by + cz < \mu + \omega \quad (2.4)$$

Ce plan discret se note $\mathcal{P}(a, b, c, \mu, \omega)$.

Comme en 2D, l'épaisseur arithmétique permet de contrôler la topologie du plan discret. Ainsi nous pouvons définir :

Définition 2.10 (Plan naïf et plan standard) *Soit $\mathcal{P}(a, b, c, \mu, \omega)$ un plan discret arithmétique. Nous avons :*

- *si $\omega = \max(|a|, |b|, |c|)$, \mathcal{P} est alors un plan discret naïf, ce plan est 18-connexé et il ne possède pas de trous 6-connexes (ANDRÈS, 1993) (figure 2.30–(a)) ;*
- *si $\omega = |a| + |b| + |c|$, \mathcal{P} est alors un plan discret standard (figure 2.30–(b)).*

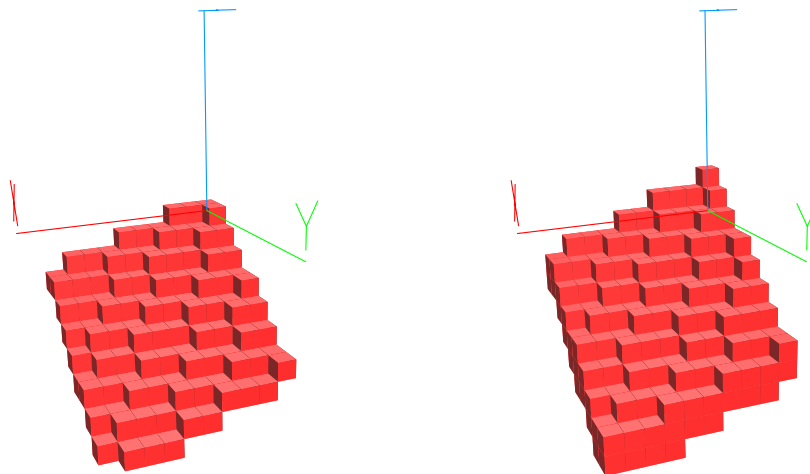


FIG. 2.30 –: (a) *plan arithmétique naïf $\mathcal{P}(6, 13, 17, 17)$ et (b) plan arithmétique standard $\mathcal{P}(6, 13, 17, 36)$ pour $0 \leq x, y \leq 10$.*

Dans ce qui suit, nous nous intéressons uniquement aux plans naïfs. Nous pouvons faire le lien entre ce plan discret et celui introduit par KIM basé sur une discrétisation GIQ :

Proposition 2.6 (DEBLED-RENNESON (1995); VITTONI (1999)) *Soit P un plan euclidien rationnel d'équation $z = -\frac{ax+by+\mu}{c}$ avec $0 \leq a \leq b \leq c$ et $c \neq 0$, nous avons :*

- la discrétisation GIQ de P (équation 2.3) coïncide avec le plan naïf $\mathcal{P}(a, b, c, \mu + \lceil \frac{c}{2} \rceil)$
- la discrétisation OBQ de P coïncide avec le plan naïf $\mathcal{P}(a, b, c, \mu)$

Nous pouvons caractériser, comme pour les droites discrètes arithmétiques, des éléments importants comme les plans d'appui. Tout d'abord, nous appelons **réseau d'indice k** l'ensemble des voxels vérifiant :

$$ax + by + cz = k$$

Les voxels vérifiant cette équation sont solutions d'une équation linéaire diophantienne. Il existe donc de nombreuses techniques pour trouver une base de cet ensemble permettant de générer tous les voxels solutions. Par exemple, REVEILLÈS (2001) propose l'utilisation de l'algorithme de BLANKINSHIP (1963).

Dans une représentation du plan par restes, un voxel appartenant au réseau d'indice k sera associé au **reste k** . Plus formellement, pour un plan discret $\mathcal{P}(a, b, c, \mu, \omega)$, la fonction des restes, notée $R(a, b, c)(x, y, z)$ est définie par (voir figure 2.31) :

$$R(a, b, c)(x, y, z) = ax + by + cz$$

Proposition 2.7 *Le plan discret $\mathcal{P}(a, b, c, \mu, \omega)$ est :*

- l'union des réseaux d'indices k pour k dans $[\mu, \mu + \omega[$
- l'ensemble des voxels de restes compris dans $[\mu, \mu + \omega[$

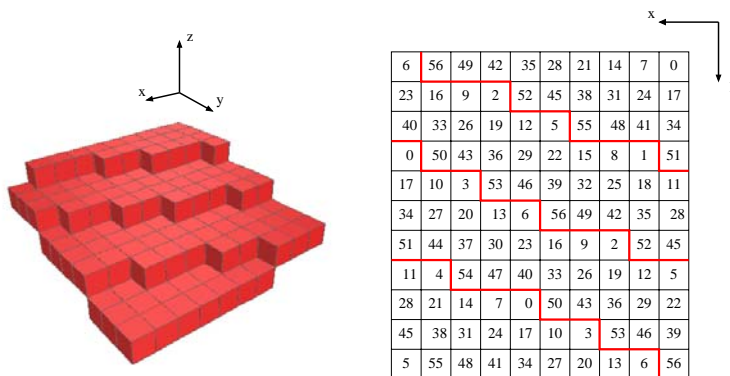


FIG. 2.31 – Morceau de plan discret $\mathcal{P}(7, 17, 57, 0)$ et représentation par reste pour $0 \leq x, y \leq 10$.

Comme en 2D, nous appelons **plan d'appui supérieur** l'ensemble des voxels du réseau d'indice μ , ces voxels vérifient donc :

$$ax + by + cz = \mu$$

Nous appelons aussi **plan d'appui inférieur** l'ensemble des voxels du réseau d'indice $\mu + \omega - 1$. En d'autres termes, ces voxels vérifient :

$$ax + by + cz = \mu + \omega - 1$$

Les voxels du plan d'appui supérieur (resp. plan d'appui inférieur) sont appelés **points d'appui supérieurs** (resp. **points d'appui inférieurs**).

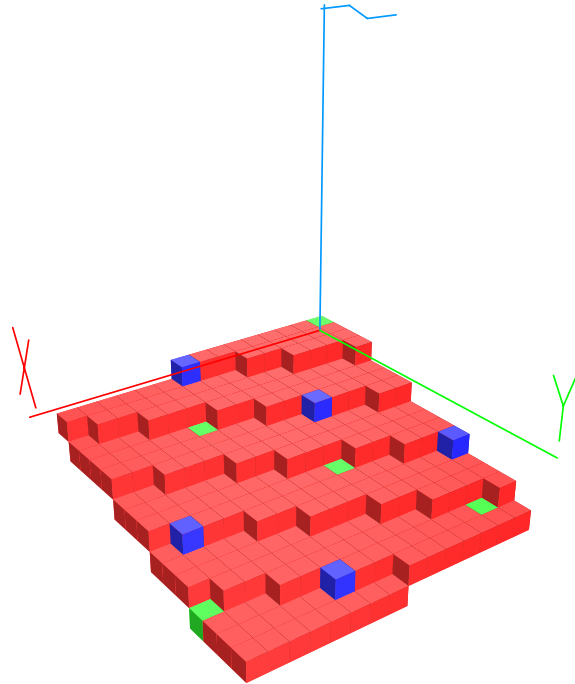


FIG. 2.32 — Illustration des points d'appui d'un plan discret : plan naïf d'équation $\mathcal{P}(7, 17, 57, 0)$ pour $0 \leq x \leq 15$ et $0 \leq y \leq 15$.

En dimension 2, la notion de périodicité des droites discrètes est assez claire avec l'ordre intrinsèque des pixels de ces courbes. En 3D, les plans discrets présentent aussi de telles structures. Dans le cas d'un plan naïf, nous avons :

$$R(a, b, c)(x, y + \kappa c, z + \lambda c) = R(a, b, c)(x, y, z)$$

pour κ, λ dans \mathbb{N} .

Le reste des voxels d'un plan discret a donc une structure bi-périodique (DEBLED-RENNESON, 1995). Ainsi, de manière intuitive, si nous regardons le voisinage d'un voxel de reste k du plan discret, ce voisinage sera identique pour tout autre voxel de même reste.

Plus formellement, les voxels d'un plan naïf sont formés par une base de deux vecteurs \vec{u} et \vec{v} liés aux paramètres du plan :

Théorème 2.9 (DEBLED-RENNESON (1995)) Une base \mathcal{B} du plan $\mathcal{P}(a, b, c, \mu)$ est composée des vecteurs $u(u_x, u_y, u_z)$ et $v(v_x, v_y, v_z)$ vérifiant :

$$\begin{aligned} |c| &= \begin{vmatrix} u_x & v_x \\ u_y & v_y \end{vmatrix} \\ |b| &= \begin{vmatrix} u_x & v_x \\ u_z & v_z \end{vmatrix} \\ |a| &= \begin{vmatrix} u_y & v_y \\ u_z & v_z \end{vmatrix} \end{aligned}$$

En plus de cette périodicité, nous pouvons caractériser plus précisément la structure des plans discrets. En effet, si nous considérons toutes les configurations possibles de voxels dans un voisinage de taille $n \times m$, nous pouvons complètement écrire la grammaire d'apparition de ces formes géométriques en fonction des paramètres du plan discret considéré. On parle alors de (n, m) -cubes (DEBLED-RENNESON, 1995; VITTONI et CHASSERY, 1997; VITTONI, 1999).

Enfin nous terminons notre analyse des plans discrets par la structure des droites discrètes présentes dans ceux-ci (voir figure 2.33). En effet, nous avons la propriété suivante :

Théorème 2.10 (DEBLED-RENNESON (1995)) *Soit $\mathcal{P}(a, b, c, \mu)$ un plan discret naïf, l'intersection de celui-ci avec le plan :*

- $z = z_0$, la projection sur le plan Oxy est la droite discrète épaisse $\mathcal{D}(a, b, \mu - cz_0, c)$;
- $y = y_0$, la projection sur le plan Oxz est la droite discrète naïve $\mathcal{D}(a, c, \mu - by_0, c)$;
- $x = x_0$, la projection sur le plan Oyz est la droite discrète naïve $\mathcal{D}(b, c, \mu - ax_0, c)$.

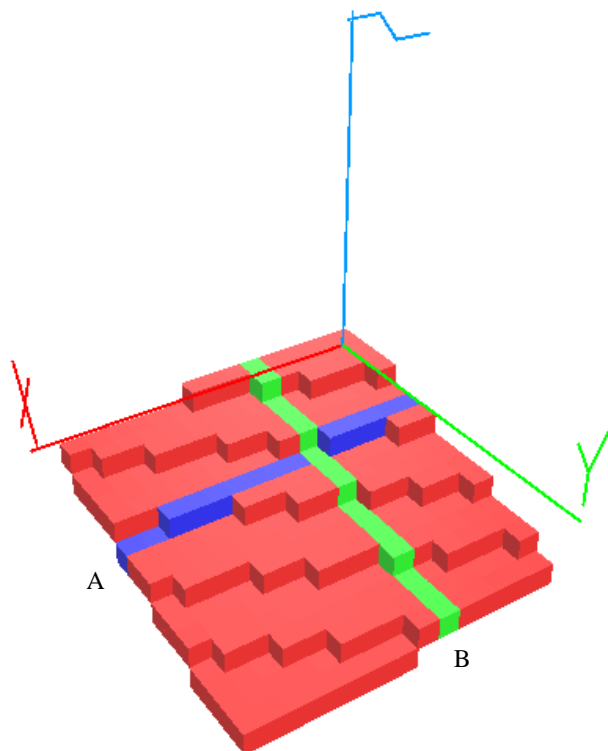


FIG. 2.33 –: *Présence de droites discrètes dans des plans naïfs : plan discret naïf $\mathcal{P}(7, 17, 57, 0)$ et droites naïves $\mathcal{D}(7, 57, -102)$ (A) et $\mathcal{D}(17, 57, -42)$ (B).*

2.4.2 Reconnaissance et facettisation d'objets discrets

Nous nous intéressons maintenant au problème de reconnaissance d'un plan discret. Comme en 2D, nous pouvons distinguer deux classes d'algorithmes. La première permet de tester la *coplanarité* d'un ensemble de voxels. En d'autres termes, ces algorithmes implémentent un prédicat qui permet de décider si un ensemble de voxels est un morceau de plan discret ou non. La seconde classe s'intéresse à une caractérisation des plans solutions. Ainsi, comme en dimension 2, nous souhaitons obtenir soit la *pré-image* du plan discret, c'est-à-dire l'ensemble des plans euclidiens se discrétisant dans l'ensemble de voxels considérés, soit une paramétrisation arithmétique de celui-ci.

2.4.2.1 Approches de KIM, VEELAERT et DEBLED-RENNESON

Dans un premier temps, nous présentons l'approche basée sur le théorème 2.8 de KIM (1984b). Une implémentation directe de ce théorème se déroulerait en trois étapes (KIM, 1984b) :

1. vérifier les hypothèses émises par KIM sur l'ensemble de voxels (projection bijective sur un des plans d'axe, projection convexe...) ;
2. construire l'enveloppe convexe des voxels ;
3. s'il existe une face de l'enveloppe convexe vérifiant le théorème 2.8, l'ensemble de voxels est un morceau de plan.

Sous cette forme, cet algorithme n'est pas très efficace à cause du test exhaustif des facettes de l'enveloppe convexe. Plus formellement, si nous testons la coplanarité d'un ensemble de n voxels, cet algorithme est en $O(n^4)$.

Une optimisation a cependant été proposée par KIM et STOJMENOVIC (1991) permettant un parcours plus efficace des facettes testées. La complexité de ce dernier algorithme se ramène donc à $O(n^2 \log(n))$.

En énonçant la propriété de régularité des plans discrets (définition 2.8), VEELAERT (1994) propose un algorithme de reconnaissance de morceaux de plan discret rectangulaires ou rectangulaires partiellement étendus. Ainsi, considérant cette propriété de régularité des plans, nous obtenons finalement un algorithme en $O(n^2)$ où n est le nombre de voxels de l'ensemble reconnu.

En se basant sur la reformulation du théorème de KIM, DEBLED-RENNESON (1995) propose une reconnaissance incrémentale de plan arithmétique. Cette approche se base sur la construction du polygone d'appui inférieur (resp. supérieur) d'un plan discret qui correspond à l'enveloppe convexe des points d'appui inférieurs (resp. supérieurs). L'idée est de construire les vecteurs \vec{u} et \vec{v} de la base du plan en considérant les sommets de ces polygones d'appui (voir figure 2.34).

Le principe de cet algorithme est qu'à chaque insertion de point, nous reconstruisons, si besoin, ces polygones d'appui pour ensuite reconstruire la base du plan reconnu. DEBLED-RENNESON (1995) a montré la complexité de cette approche pour résoudre tous les cas pathologiques permettant une reconstruction arithmétique. En effet, certains points de cet algorithme se base sur des conjectures dont certaines ont été récemment résolues par MESMOUDI (2002).

Dans le paragraphe suivant, nous détaillons un peu plus les approches basées sur une analyse de la pré-image, dans l'espace des paramètres, associée aux plans discrets.

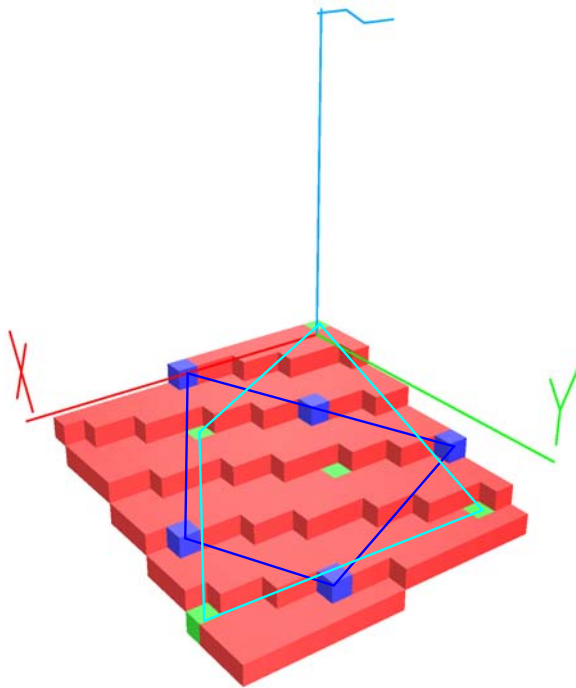


FIG. 2.34 –: Illustration des polygones d’appui pour un morceau de plan naïf $\mathcal{P}(7, 17, 57, 0)$.

2.4.2.2 Approche basée sur la structure du dual

Considérons un ensemble \mathcal{E} de voxels. Si cet ensemble de voxels forme un plan discret naïf dans le premier 48^{ème} d’espace (*i.e.* $x \geq y \geq z > 0$), il existe alors un plan euclidien dont la discrétisation contient \mathcal{E} . Plus formellement, il existe un triplet (α, β, γ) dans $[0, 1]^2 \times [0, 1[$ tel que \mathcal{E} soit contenu dans l’ensemble :

$$\mathcal{P} = \{(x, y, z) \in \mathbb{Z}^3 \mid 0 \leq \alpha x + \beta y + \gamma + z < 1\}$$

Si nous voulons construire l’ensemble des plans se discrétisant dans l’ensemble \mathcal{E} , nous définissons l’ensemble $\bar{\mathcal{E}}$ dans l’espace des paramètres défini par (voir figure 2.35²) :

$$\bar{\mathcal{E}} = \{(\alpha, \beta, \gamma) \in [0, 1]^2 \times [0, 1[\mid \forall (x, y, z) \in \mathcal{E} \ 0 \leq \alpha x + \beta y + \gamma + z < 1\} \quad (2.5)$$

Comme en dimension 2, la construction de ce domaine ou simplement le test d’existence d’une solution se ramène à un problème classique de programmation linéaire.

Nous pouvons, par exemple, utiliser l’algorithme de MEGIDDO (1984) (théorème 2.2) qui nous permet d’avoir un algorithme linéaire en le nombre de points pour tester la coplanarité d’un ensemble de voxels. De plus, nous disposons de l’amélioration de cet algorithme proposée par BUZER (2002) qui nous permet d’avoir un test de coplanarité incrémental en $O(n)$ où n est le nombre de voxels dans \mathcal{E} ce qui est optimal pour le problème.

2. Merci à Isabelle Sivignon du LIS (Grenoble) pour son aide sur la construction de ces pré-images avec l’algorithme de VITTONÉ (1999).

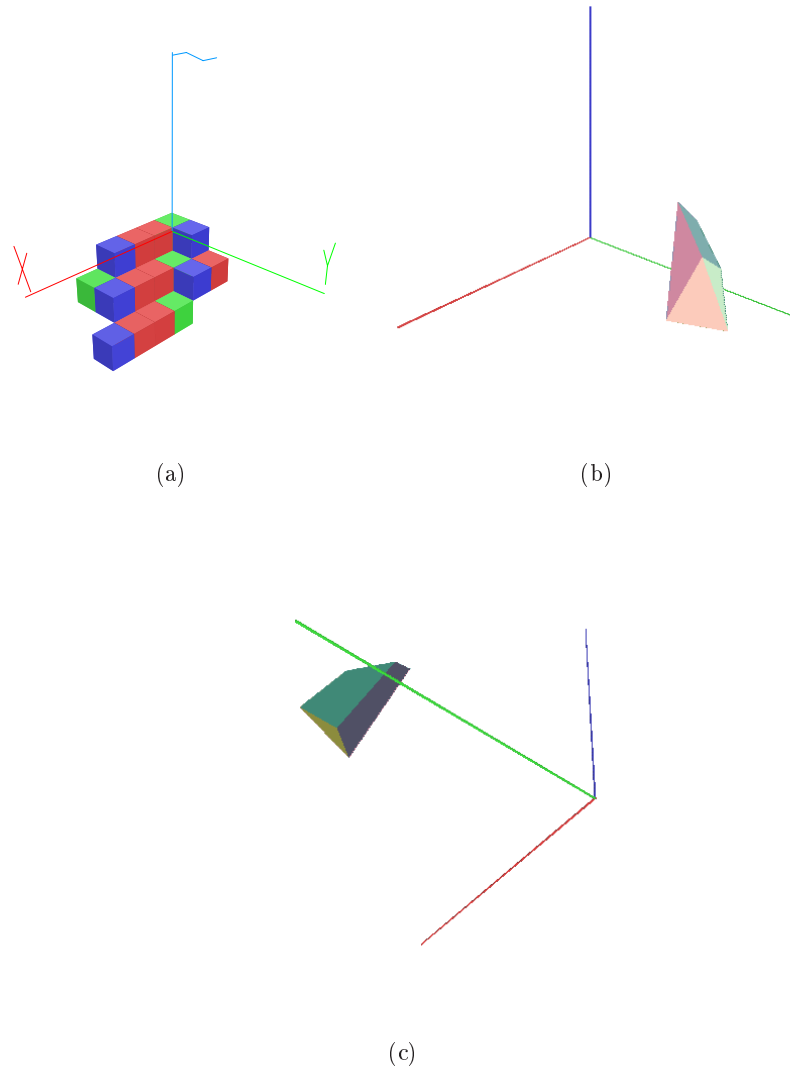


FIG. 2.35 —: (a) Morceau de plan $P(1,3,4,0)$ avec ses points d'appui et (b)-(c) différentes vues du polyèdre des plans euclidiens $\bar{\mathcal{E}}$ dont la discrétisation contient le morceau P .

Si nous souhaitons décrire entièrement le polyèdre des solutions, FRANÇON ET AL. (1996) proposent l'utilisation de la réduction de FOURIER-MOTZKIN pour construire le polyèdre $\bar{\mathcal{E}}$ mais la complexité, très coûteuse, de cet algorithme rend son utilisation très peu avantageuse pour la reconnaissance de plan.

Afin d'avoir une écriture des sommets du polyèdre sous forme de fractions irréductibles, VITTONÉ (1999) propose une utilisation de l'algorithme de GRABINER (1992). Ainsi, les sommets sont sous la forme $(\frac{a}{c}, \frac{b}{c}, \frac{\mu}{c})$ (dans le premier 48^{ème} d'espace) permettant ainsi l'extraction d'une paramétrisation arithmétique naïve du morceau reconnu donnée par $\mathcal{P}(a, b, c, \mu)$.

L'idée est similaire au cas 2D, à chaque insertion d'une contrainte dans le système (c'est-à-dire à chaque ajout de voxel), nous regardons l'ensemble des arêtes coupées par celle-ci. Chaque intersection engendre donc un nouveau sommet du polyèdre solution dont les coordonnées, sous forme de fractions irréductibles, sont calculées en appliquant l'algorithme de GRABINER (1992) sur les coordonnées des extrémités de l'arête coupée (voir algorithme 3).

L'algorithme initial de VITTONÉ (1999) est peu efficace à cause du test exhaustif sur toutes les arêtes du polyèdre introduisant ainsi une complexité en $O(n^3 \log(n))$. Nous pouvons cependant baisser cette borne asymptotique en utilisant des algorithmes optimaux, classiques en géométrie algorithmique ou en programmation linéaire, couplés avec l'algorithme de réduction de GRABINER. Ainsi, en utilisant une extension tridimensionnelle de l'algorithme de PREPARATA et SHAMOS (1985), et en appliquant la réduction de GRABINER, nous pouvons proposer une borne en $O(n \log^2(n))$. Nous ne détaillons pas cet algorithme car celui-ci est très similaire à l'algorithme 4.

2.4.2.3 Vers une structure complète du domaine dans l'espace dual

Lors de l'analyse des droites discrètes dans l'espace dual, des théorèmes très puissants ont été démontrés permettant de connaître exactement la structure de la pré-image dans l'espace des paramètres associée à un segment discret. De plus, cette analyse nous a permis d'écrire des algorithmes optimaux pour le problème de la reconnaissance de ces objets.

Dans le cas tridimensionnel, des solutions algorithmiques très efficaces, pour le test de coplanarité ou la construction du domaine dans le dual, existent en programmation linéaire (BUZER, 2002; PREPARATA et SHAMOS, 1985). Cependant très peu de solutions exploitent pleinement la structure très particulière des plans discrets. Ainsi, de nombreuses questions nous semblent intéressantes : en imposant une contrainte de connexité de l'ensemble \mathcal{E} , peut-on réduire la borne $O(n \log(n))$ proposée par PREPARATA et SHAMOS ? Quel est le nombre de faces du polyèdre des solutions ? Est-ce que ce polyèdre possède une structure arithmétique ?

Dans cette analyse, nous montrons que ce polyèdre possède une structure particulière qu'il serait dommage de ne pas prendre en compte dans les algorithmes de reconnaissance de plans discrets.

Dans un premier temps, nous montrons qu'il existe deux sommets particuliers du polyèdre qui correspondent aux plans d'appui d'un plan discret :

Proposition 2.8 *Étant donné un morceau de plan naïf $\mathcal{P}(a, b, c, \mu)$, le polyèdre $\bar{\mathcal{E}}$ des plans euclidiens dans l'espace des paramètres contient deux sommets particuliers :*

- $(\frac{a}{c}, \frac{b}{c}, \frac{\mu}{c})$ qui correspond au plan d'appui inférieur de \mathcal{P} ;

- $\left(\frac{a}{c}, \frac{b}{c}, \frac{\mu+1}{c}\right)$ qui correspond au plan d'appui supérieur de \mathcal{P} .

De plus, nous avons :

- les faces du polyèdre adjacentes au point $\left(\frac{a}{c}, \frac{b}{c}, \frac{\mu}{c}\right)$ correspondent aux sommets du polygone d'appui inférieur ;
- les faces du polyèdre adjacentes au point $\left(\frac{a}{c}, \frac{b}{c}, \frac{\mu+1}{c}\right)$ correspondent aux sommets du polygone d'appui supérieur.

Preuve : Considérons tout d'abord un morceau connexe de plan discret $\mathcal{P}(a, b, c, \mu)$ dans le premier 48^{ème} d'espace. L'ajout du voxel (x, y, z) correspond à l'introduction de deux demi-espaces dans l'espace des paramètres :

$$\begin{aligned} C_1 : \quad & \alpha x + \beta y + \gamma + z \geq 0 \\ C_2 : \quad & \alpha x + \beta y + \gamma + z < 1 \end{aligned}$$

De la même manière que l'algorithme de PREPARATA et SHAMOS (1985), nous traitons indépendamment les contraintes C_1 et C_2 . Concernant les contraintes C_2 , nous souhaitons calculer l'enveloppe inférieure de celles-ci et sur les contraintes C_1 , nous considérons l'enveloppe supérieure. Étant donné que les voxels appartiennent à un plan discret, l'union de ces deux enveloppes sera non vide et définira exactement le polyèdre des plans euclidiens se discrétisant dans P , dans l'espace des paramètres. Dans ce qui suit, nous considérons les contraintes C_2 sans l'inégalité stricte pour simplifier les explications. Le polyèdre final s'obtiendra en excluant toutes les faces et sommets issus des contraintes C_2 .

Si nous considérons uniquement les contraintes C_2 , celles-ci correspondent au passage dans l'espace dual des points (x, y, z) . Les contraintes C_1 correspondent au passage dans le dual des points $(x, y, z + 1)$.

Soit P_{sup} le plan d'appui supérieur associé au morceau de plan \mathcal{P} . Par définition de ce plan, tous les voxels (x, y, z) de \mathcal{P} sont sous ce plan (dans le premier 48^{ème} d'espace). Ainsi, dans l'espace dual, toutes les contraintes C_2 des points (x, y, z) contiennent le point P_{sup}^* transformé de P_{sup} . De la même manière, toutes les contraintes C_1 , dans l'espace dual, associées aux points $(x, y, z + 1)$ contiennent le point P_{inf}^* correspondant au plan d'appui inférieur P_{inf} traduit de 1 en z .

De plus, si nous considérons les contraintes C_2 associées aux points d'appui supérieurs, celles-ci passent par le point P_{sup}^* (par définition des points d'appui, ceux-ci sont les seuls points de \mathcal{P} contenus dans P_{sup}). Étant donné que chaque contrainte partitionne l'espace des paramètres, le point P_{sup}^* est donc un sommet du polyèdre final associé au morceau de plan \mathcal{P} . De plus les faces adjacentes correspondent à des points d'appui supérieurs de \mathcal{P} . De même, le point P_{inf}^* est un sommet du polyèdre final associé au morceau de plan \mathcal{P} et les faces adjacentes à ce point correspondent à des points d'appui inférieurs \mathcal{P} .

Nous pouvons affiner encore notre analyse en ne considérant que les points d'appui sommets des polygones d'appui. En effet, un voxel $v(v_x, v_y, v_z)$ intérieur au polygone d'appui supérieur engendre une face adjacente à P_{sup}^* dont la normale est donnée par $(v_x, v_y, v_z)^T$. Ce vecteur normal est orthogonal à la face et sa direction est contraire à celle de l'inégalité.

Si nous notons $\{e^i\}_{i=1..N}$ les sommets du polygone d'appui supérieur et si v est intérieur à ce polygone, nous avons :

$$v = \sum_{i=1}^N w^i e^i$$

avec $\{w^i\}_{i=1..N}$ dans \mathbb{R}^3 et $w_j^i > 0$ pour $1 \leq i \leq N$ et $1 \leq j \leq 3$.

En d'autres termes, la normale du plan engendré par v est un barycentre à poids positifs, notés w_j^i , des normales issues des sommets du polygone d'appui supérieur. Ainsi, la contrainte issue de v ne crée donc pas de face au polyèdre (voir figure 2.36).

Un raisonnement similaire permet de montrer que seuls les sommets du polygone d'appui inférieur engendrent des facettes adjacentes à P_{inf}^* .

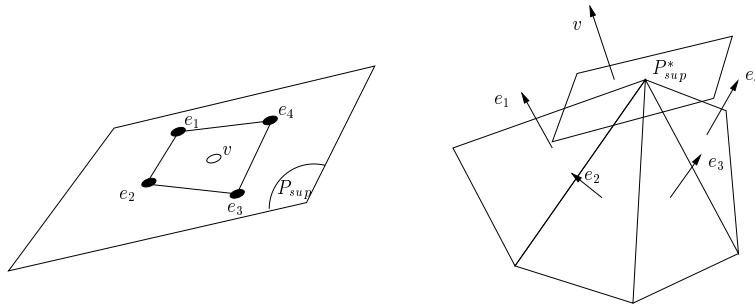


FIG. 2.36 – : Illustration de la contrainte engendrée par un point d'appui v contenu dans le polygone d'appui de sommets notés $\{e^i\}_{i=1..4}$.

□

Nous connaissons maintenant certaines propriétés de la géométrie du polyèdre : nous avons deux sommets particuliers ayant la même abscisse et la même ordonnée avec une hauteur en γ qui ne diffèrent que de $\frac{1}{c}$ et qui correspondent aux plans d'appui du morceau de plan discret. De plus, nous connaissons exactement la géométrie des faces adjacentes à ces sommets. D'une manière informelle, la figure 2.37 illustre notre connaissance actuelle du polyèdre : nous connaissons deux sommets et les faces adjacentes à ceux-ci. Par contre, nous ne connaissons pas encore ce qu'il se passe à l'intersection de ces cônes.

Avant de continuer, nous pouvons encore affiner notre analyse actuelle en dénombrant les facettes adjacentes aux sommets P_{sup}^* ou P_{inf}^* . Nous avons montré que ce nombre est égal au nombre de sommets de l'enveloppe convexe des points d'appui. Or, ces points d'appui ont une structure arithmétique particulière puisqu'ils appartiennent à un réseau défini par deux vecteurs solutions d'une équation diophantienne $ax + by + cz = r$.

Considérons un réseau donné par (voir figure 2.38) :

$$\left(i, \left\{ \frac{ai}{b} \right\} \right) \quad \text{pour } 0 < i < b$$

REVEILLÈS et YAACOUB (1995) ont montré que le calcul de l'enveloppe convexe d'un tel réseau est lié au développement en fraction continue de $\frac{a}{b}$. Ils ont ensuite

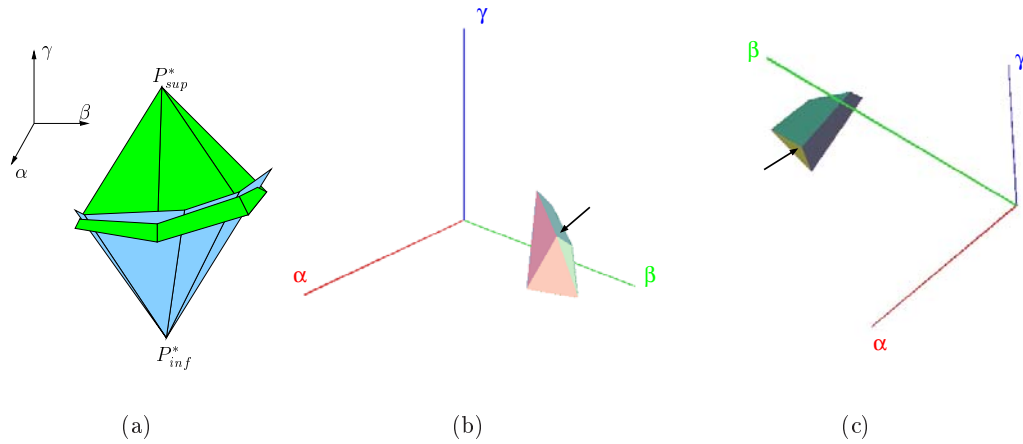


FIG. 2.37 — Illustration du polyèdre avec la notion de double-cône : (a) illustration schématique du double-cône et (b)-(c) le polyèdre dans l'espace dual du morceau de plan $P(1, 3, 4, 0)$ (voir figure 2.35), les flèches indiquent respectivement les sommets P_{sup}^* et P_{inf}^* .

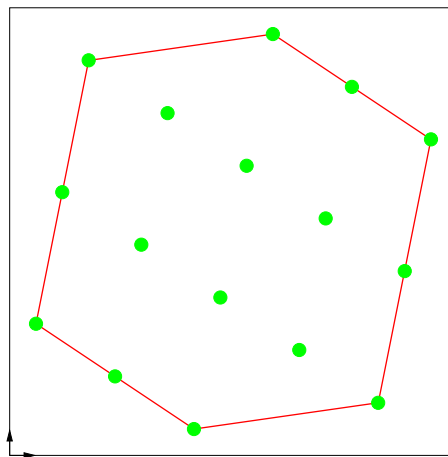


FIG. 2.38 — Enveloppe convexe du réseau engendré par la suite $\{(i, \{\frac{ai}{b}\})$ avec $0 < i < b\}$ pour $a = 5$ et $b = 17$.

proposé un algorithme, dont la complexité est en $O(\log(a))$, permettant de construire une telle enveloppe convexe dont le nombre de sommets est aussi borné par $O(\log(a))$.

Ce cas de figure est idéal dans le sens où nous n'avons qu'une seule période du réseau. Cependant, en considérant un morceau de plan quelconque, cette théorie devrait nous permettre de borner ce nombre de faces avec des bornes assez intéressantes par rapport au nombre de voxels. D'une manière générale, nous pouvons utiliser la borne de ACKETA et ŽUNIĆ (1995) sur le nombre maximal d'arêtes d'un polygone discret convexe dans une grille $n \times n$ qui est en $O(n^{2/3})$ (voir paragraphe 3.3.5).

Si nous revenons maintenant à notre analyse du polyèdre, nous pouvons prouver le théorème suivant :

Théorème 2.11 *Étant donné un morceau de plan discret \mathcal{P} dans le premier 48^{ème} d'espace dont les voxels se projettent, sur le plan Oxy , à l'intérieur des projections des polygones d'appui. Le polyèdre $\tilde{\mathcal{E}}$ dans l'espace des paramètres est exactement l'intersection des deux cônes issus de P_{sup}^* et P_{inf}^* .*

Preuve : *Considérons un voxel v dont la contrainte C_2 associée à v , notée $C_2(v)$, ne passe pas par P_{sup}^* (v n'est donc pas un point d'appui). Supposons que la contrainte $C_2(v)$ engendre une nouvelle face au polyèdre, en d'autres termes, supposons que $C_2(v)$ appartient à l'enveloppe inférieure des contraintes C_2 (voir figure 2.39-(a) pour une illustration en 2D).*

La contrainte $C_2(v)$ coupe la droite parallèle à l'axe γ passant par P_{sup}^ au point p . Sur cette droite, p est nécessairement au-dessus de P_{sup}^* sinon ce dernier ne serait pas sommet du polyèdre (voir figure 2.39-(b)).*

La projection verticale du voxel v sur le plan P_{sup} dans l'espace primal correspond dans l'espace dual à une translation de la contrainte $C_2(v)$ de vecteur $\overrightarrow{pP_{sup}^}$. Si $C_2(v)$ fait partie de l'enveloppe inférieure des contraintes, la contrainte issue de la translation de vecteur $\overrightarrow{pP_{sup}^*}$ est aussi dans l'enveloppe inférieure des contraintes (figure 2.39-(c)).*

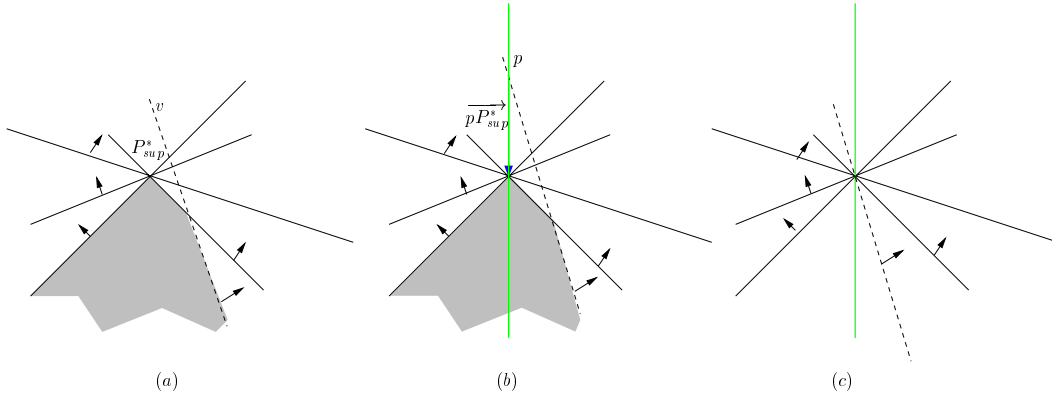


FIG. 2.39 — *Illustration dans le cas 2D de la preuve du théorème 2.11 : (a) la contrainte $C_2(v)$ ne passe pas par P_{sup}^* mais engendre une face du polyèdre, (b) et (c) translation de la contrainte du vecteur $\overrightarrow{pP_{sup}^*}$.*

Si v se projette verticalement sur P_{sup} à l'intérieur du polygone d'appui, la normale associée à la contrainte $C_2(v)$ est un barycentre à poids positifs des normales associées aux sommets du polygone d'appui (nous rappelons que le

vecteur normal est orthogonal à la face et sa direction est contraire à celle de l'inégalité). Cela implique que la contrainte $C_2(v)$ traduite de $\overrightarrow{pP_{sup}^*}$ ne fait pas partie de l'enveloppe inférieure des contraintes et donc la contrainte $C_2(v)$ n'engendre pas de face supplémentaire au polyèdre.

Finalement, une condition nécessaire pour que les contraintes C_2 associées à un morceau de plan \mathcal{P} n'engendent pas de faces au polyèdre, est que les voxels associés se projettent sur le plan Oxy dans la projection du polygone d'appui supérieur.

En utilisant le même raisonnement pour les contraintes C_1 , si le morceau de plan \mathcal{P} dans le premier 48^{ème} d'espace est tel que ses voxels se projettent sur le plan Oxy , à l'intérieur des projections des polygones d'appui (voir figure 2.40), alors le polyèdre des solutions est exactement l'union des deux cônes issus des points P_{sup}^* et P_{inf}^* .

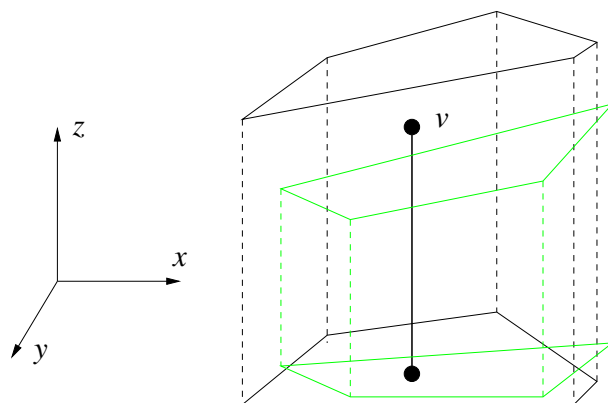


FIG. 2.40 — Condition nécessaire pour que le polyèdre des solutions dans l'espace des paramètres soit un double-cône : chaque voxel doit se projeter sur le plan Oxy , dans les projections des polygones d'appui.

□

Grâce à ce théorème, nous pouvons calculer le nombre de facettes du polyèdre $\bar{\mathcal{E}}$:

Corollaire 2.1 Soit $\bar{\mathcal{E}}$ le polyèdre dans l'espace des paramètres associé à un morceau de plan P vérifiant les hypothèses du théorème 2.11. Le nombre de faces de ce polyèdre est le nombre de sommets des deux polygones d'appui de P .

Remarquons que dans les hypothèses du théorème, les voxels de \mathcal{P} n'ont aucune autre contrainte que celle de la projection dans les polygones d'appui. \mathcal{P} peut donc être déconnecté ou avoir des trous, la forme du domaine est identique.

Dans les analyses expérimentales que nous avons effectuées, tous les polyèdres $\bar{\mathcal{E}}$ avaient cette structure en double-cône, même dans les cas où les hypothèses du théorème n'étaient pas vérifiées. Ceci nous permet d'énoncer la conjecture dans laquelle la seule contrainte sur P est qu'il soit connexe :

Conjecture 2.1 Soit $\mathcal{P}(a, b, c, \mu)$ un morceau de plan connexe, le polyèdre dans l'espace des paramètres associé à \mathcal{P} a la forme du double-cône, c'est-à-dire :

- il existe deux sommets caractéristiques de coordonnées $(\frac{a}{c}, \frac{b}{c}, \frac{\mu}{c})$ et $(\frac{a}{c}, \frac{b}{c}, \frac{\mu+1}{c})$;

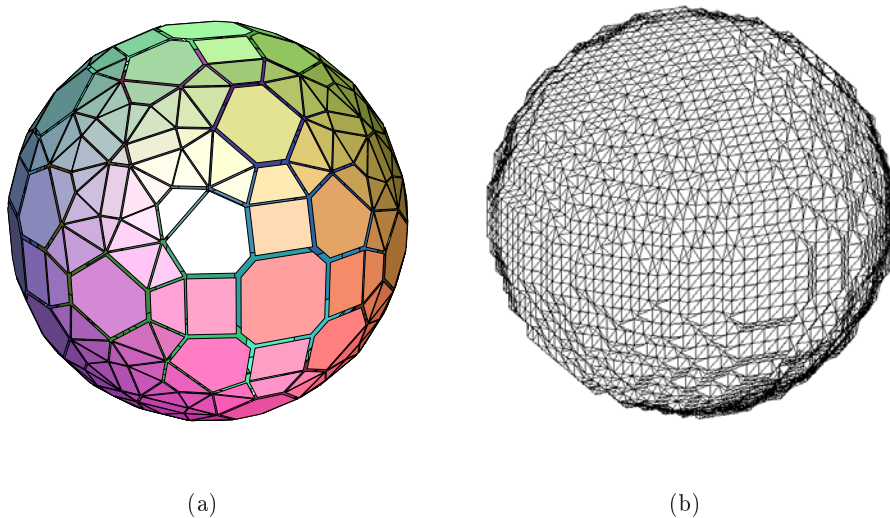


FIG. 2.41 –: Facettisation basée sur l’enveloppe convexe 3D (a) et sur un algorithme de type *Marching-Cubes* (b).

- les faces du polyèdre correspondent uniquement aux sommets des polygones d’appui de \mathcal{P} .

Dans le paragraphe suivant, nous nous intéressons à une application classique de ces algorithmes de reconnaissance de plan qui consiste en la facettisation d’un objet discret.

2.4.2.4 Facettisation d’objets discrets

Dans de nombreuses applications, la manipulation ou simplement la visualisation d’objets discrets 3D est assez coûteuse en temps. Ainsi, la facettisation ou polyédrisation de ces objets discrets apparaît comme outil primordial.

L’objectif est de construire une approximation polyédrique du volume discret avec les contraintes suivantes : *les sommets du polyèdre doivent être des points discrets et l’approximation doit être réversible*. En d’autres termes, cette facettisation ne doit perdre aucune information et le volume discret doit pouvoir être reconstruit entièrement à partir de celle-ci.

Dans le cas d’objets 2D, les algorithmes de segmentation en droite discrète offrent une solution très efficace à ce problème. Dans le cas tridimensionnel, les choses sont plus complexes comme nous le verrons.

Tout d’abord, nous présentons deux techniques de facettisation réversibles que l’on peut qualifier de *naïves*. La première n’est valide que pour les volumes discrets convexes (qui sont la discrétisation d’objets euclidiens convexes) et consiste en un calcul de l’enveloppe convexe 3D (voir figure 2.41-(a)). La facettisation s’obtient en utilisant un algorithme classique en géométrie algorithmique dont il existe de nombreuses implémentations très optimisées (outil `qhull`³ par exemple). Cette polyédrisation n’est bien évidemment réversible que dans le cas d’objets convexes.

3. <http://www.geom.umn.edu/software/qhull>

Une seconde approche *naïve* se base sur les algorithmes de type *Marching-Cubes* (LORENSEN et CLINE, 1987; KENMOCHI ET AL., 1996). Dans ce cas, la facettisation s'effectue à l'aide d'une table de correspondance entre configurations $2 \times 2 \times 2$ de voxels et les ensembles de facettes associés à ces configurations (voir figure 2.41-(b)). Ainsi, en parcourant la surface discrète, chaque configuration de voxel est remplacée par un morceau de surface triangulée. Bien évidemment, la table de correspondance est construite de manière à garantir un certain nombre de propriétés topologiques sur cette surface (LACHAUD, 1996).

Par construction de la table de correspondance, ce processus est bien évidemment réversible mais le nombre de facettes peut être très important même pour des objets simples comme des plans discrets. Dans un objectif de visualisation, des techniques de simplification ou de décimation de surfaces sont alors nécessaires (voir par exemple COEURJOLLY ET AL. 1999). Une fois encore, des implémentations très efficaces existent en séquentiel ou sur machines parallèles (MIGUET et NICOD, 1997). En plus de leur simplicité, ces deux approches ont un avantage très important : la facettisation est unique.

En utilisant des objets classiques de la géométrie discrète, il est tout à fait normal de considérer une facettisation basée sur la notion de plan discret. Ainsi, un tel algorithme de facettisation doit contenir les caractéristiques suivantes :

- un algorithme de reconnaissance de plan discret ;
- un processus de placement de *germes* de manière parallèle ou incrémentale ;
- à partir d'un germe, un processus d'ajout de voxels adjacents afin de construire les morceaux de plans discrets.

Une des premières approches fut proposée par BORIANNE et FRANÇON (1994) mais le processus de reconnaissance de plan se basait sur une analyse au moindre carré ce qui rendait le processus non discret.

Par la suite, DEBLED-RENNESON (1995) propose une approche basée sur son algorithme de reconnaissance de plans discrets naïfs mais uniquement pour des objets symétriques par rapport à un $48^{\text{ème}}$ d'espace. La polyédrisation se fait incrémentalement, c'est-à-dire qu'un premier germe est choisi sur la surface et à partir de celui-ci, un morceau de plan discret maximal est calculé. Une fois la reconnaissance terminée, un autre germe est choisi sur la surface (adjacent au morceau de plan reconnu) et le processus continue tant que tous les voxels n'ont pas été visités. Le parcours utilisé se fait ligne par ligne pour un y fixé.

Par la suite, FRANÇON et PAPIER (1999); PAPIER (1999) proposent un algorithme séquentiel basé sur une reconnaissance de plan par l'algorithme de FOURIER-MOTZKIN adapté aux plans discrets (FRANÇON ET AL., 1996). Étant donné un germe choisi sur la surface, le mode de parcours est une propagation *topologique* (*i.e.* basée sur la relation d'adjacence des surfels) autour de celui-ci. En d'autres termes, un parcours en largeur du graphe d'adjacence des éléments de surface (surfels) est calculé avec un test d'appartenance au plan discret à chaque visite de surfel. Ce processus de propagation se termine soit quand il ne reste plus de surfels à visiter, soit quand la reconnaissance de plan échoue, soit enfin quand le bord du morceau de plan en cours de reconnaissance n'est plus un disque topologique.

Récemment, SIVIGNON (2001) a proposé une facettisation basée sur l'algorithme de reconnaissance de VITTONI (1999). Le processus global est lui aussi séquentiel mais l'utilisation d'un étiquetage par tricube permet de réduire le nombre de faces

généérées par rapport à l'approche de PAPIER. De plus, l'algorithme de reconnaissance étant bien plus efficace que la réduction de FOURIER-MOTZKIN, le coût global de cet algorithme est moindre.

Enfin, même si cette approche ne se base pas sur la notion de plan discret, nous pouvons citer l'algorithme de polyédrisation de BURGUET et MALGOUYRES (2000) qui consiste en une approche parallèle pour la construction des faces : dans un premier temps, des germes sont distribués sur la surface en fonction de sa courbure, les faces sont calculées en étiquetant les surfels par une propagation topologique autour de chaque germe. Dans ce cas, la facettisation n'est pas réversible étant donné qu'il n'y a aucune prise en compte de la géométrie lors de la construction des faces.

Comme nous pouvons le voir dans cette bibliographie, de nombreuses approches existent en fonction des choix d'initialisation des germes ou de processus de propagation. Nous pouvons cependant nous interroger sur le problème suivant : *Comment évaluer une facettisation ?* Dans un premier temps, si nous ne considérons pas la complexité des algorithmes, nous pouvons dire qu'un polyèdre est optimal s'il possède un nombre minimal de sommets. Par la suite, nous pourrions ajouter d'autres critères comme *l'aspect des facettes* : on dit que l'aspect des facettes d'une triangulation est *bon* si les triangles sont aussi proches que possible de triangles équilatéraux et ne sont pas dégénérés.

Considérons donc le problème du nombre minimal de sommets. Lors d'un travail récent avec ISABELLE SIVIGNON du LIS-Grenoble, nous nous sommes interrogés sur la construction d'une représentation polyédrale d'un volume discret à partir du résultat d'une facettisation quelconque. Ainsi, nous considérons le contexte général suivant : soit $E(v)$ un étiquetage de chaque voxel v avec une liste d'indices de plans discrets auxquels il appartient. À partir de cette information, nous voulons construire un graphe représentant la topologie du polyèdre final. En d'autres termes, le plongement dans \mathbb{Z}^3 des sommets de ce graphe sont les sommets de la facettisation, les arêtes du graphe seront les arêtes du polyèdre et des cycles non réductibles, les faces.

Pour une description plus détaillée de cette étude, le lecteur pourra se référer à l'article SIVIGNON et COEURJOLLY (2002). Brièvement, pour obtenir une représentation polyédrale minimale en nombre de sommets à partir de $E(v)$, nous commençons par construire un graphe d'adjacence dont les sommets sont des triplets (i, j, k) d'indices de plan discret tels qu'il existe un pointel voisin immédiat de voxels étiquetés par i, j et k . Une arête existe entre deux sommets (i, j, k) et (i', j', k') s'il existe deux indices en commun.

Pour obtenir le graphe de sommets minimaux, il nous faut réduire le graphe précédent à l'aide de deux processus de contraction : le premier consiste en une réduction des cliques de ce graphe (voir WEST 2001 pour une introduction sur les graphes). En effet, ces cliques correspondent à des intersections de plans discrets. Ainsi, en utilisant un algorithme de calcul d'une couverture d'un graphe en nombre minimal de cliques (problème du *Minimum clique covering*), nous construisons un nouveau graphe où les sommets sont ces cliques et nous plaçons une arête entre deux sommets si les deux cliques correspondantes partagent un sommet.

Ensuite, nous montrons dans SIVIGNON et COEURJOLLY (2002) qu'une seconde contraction est nécessaire. Celle-ci se base sur la construction d'un nouveau graphe dans lequel les sommets sont des cycles dans le graphe précédent. Une fois encore, pour extraire un nombre minimal de sommets, nous utilisons un processus de calcul de couverture en nombre minimal de cycles.

Finalement, nous obtenons un graphe qui correspond exactement à la représentation polyédrique de notre objet 3D avec un nombre minimal de sommets.

Bien évidemment, ce travail est avant tout théorique puisque les deux algorithmes d'extraction de cliques ou de cycles sont NP-complets (THOMASSEN, 1997; WEST, 2001). Il existe cependant des approximations polynomiales (HOCHBAUM, 1997) proches de l'optimal.

2.5 Statistique pour la reconnaissance de droites et plans discrets

Dans cette partie, nous revenons sur le problème de reconnaissance de droite et de plan discret du point de vue statistique. L'objectif ici est de mettre en place un test statistique nous permettant de reconnaître un morceau de plan discret ou un segment discret. Bien évidemment, pour ce dernier problème en 2D, nous avons présenté des algorithmes optimaux pour la reconnaissance de droite de manière exacte. Cependant, nous avons aussi illustré les difficultés pour obtenir un algorithme de reconnaissance de plan discret vraiment efficace.

Considérons le code de FREEMAN (voir figure 2.3) associé à la droite discrète naïve $\mathcal{D}(a, b, \mu)$. Dans le premier octant, ce codage contient les codes "0" et "1" tels que les "1" se trouvent *isolés* (voir détails dans le paragraphe 2.2).

Dans une approche probabiliste, nous avons la proposition suivante :

Proposition 2.9 *Considérons le processus qui consiste à choisir aléatoirement un code, sur le mot correspondant à la droite infinie $D(a, b, \mu)$ dans le premier octant. La probabilité d'obtenir le code "1" est de $\frac{a}{b}$, la probabilité d'obtenir le code "0" est de $1 - \frac{a}{b}$.*

Cette proposition se prouve directement par construction des droites discrètes.

La figure 2.42 illustre ce processus sur une droite de pente $\frac{1}{4}$: en abscisse nous avons les tirages successifs d'un code du codage et en ordonnée les fréquences d'apparition des codes "0" et "1". Nous remarquons donc que les courbes convergent vers les valeurs $\frac{1}{4}$ et $\frac{3}{4}$.

Plus formellement, le processus décrit ci-dessus suit une loi de BERNOULLI de paramètre $p = \frac{a}{b}$ qui correspond à un pile ou face avec une probabilité de p pour l'une des deux valeurs.

Sur le plan théorique, nous avons des résultats de convergence d'une telle estimation de la loi. En effet, si nous notons p le paramètre de la loi de BERNOULLI et \hat{p} le paramètre empirique estimé sur les observations (celles-ci doivent être indépendantes), nous savons que les observations de \hat{p} suivent une loi binomiale de paramètre n et p (n représente le nombre d'observations analysées). De plus, la variance de cette loi binomiale est $p(1-p)/n$, ceci signifie que l'estimation converge vers p (au sens de la moyenne quadratique) quand n tend vers l'infini. Finalement, en utilisant le *théorème central limite*, nous avons que $\sqrt{n}(\hat{p} - p)/\sqrt{p(1-p)}$ converge en loi vers une loi normale $N(0, 1)$. Ainsi, par abus de langage on dit que l'estimation de p par \hat{p} converge vers 0 en $1/\sqrt{n}$.

Si nous modifions notre processus en prenant les codes en abscisse croissante, la géométrie des droites discrètes nous permet d'avoir une vitesse de convergence très rapide vers les valeurs qui correspondent effectivement à la pente de la droite (voir figure 2.43). Le passage d'un segment à un autre correspond à un changement du

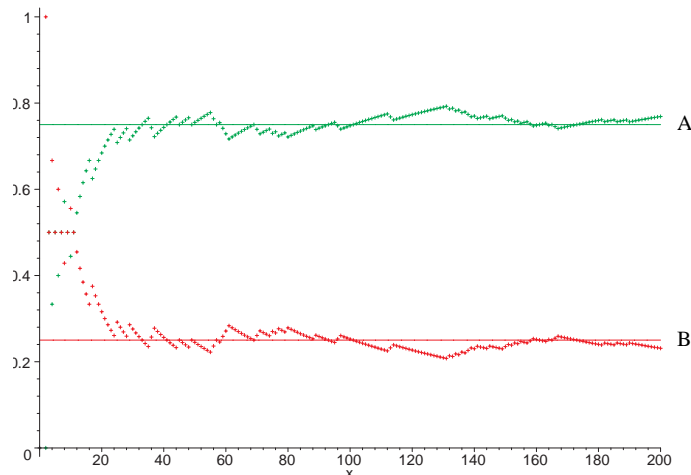


FIG. 2.42 — Liens entre droite discrète et loi de BERNOULLI : en ordonnée se trouvent les fréquences d'apparition des codes "0" (courbe A) et "1" (courbe B) lors d'un tirage aléatoire d'un code dans le mot infini issu de la droite naïve $\mathcal{D}(1, 4, 0)$. L'abscisse correspond aux tirages successifs.

paramètre p de la loi de BERNOULLI. C'est ce changement de loi que nous souhaitons détecter.

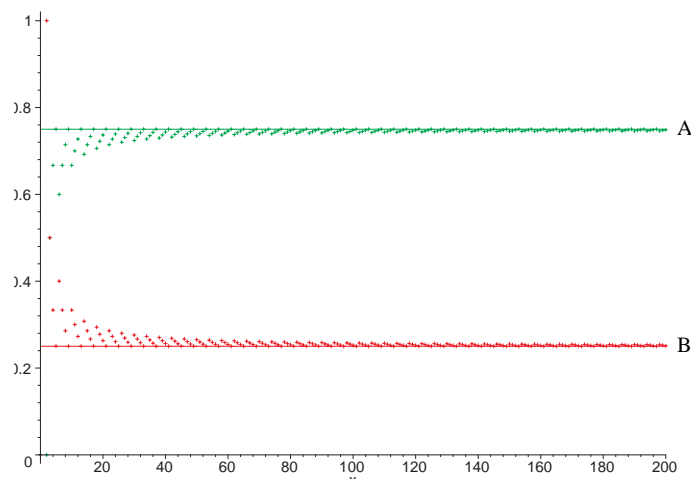


FIG. 2.43 — Fréquences d'apparition des codes quand ceux-ci sont analysés en abscisse croissante.

En se basant sur cette analyse, tout test statistique nous permettant de mesurer l'adéquation des codes du codage de FREEMAN à une loi de BERNOULLI nous permet de mettre en place un algorithme de reconnaissance de droite discrète. Encore une fois, le cas 2D n'a que peu d'intérêt mais continuons l'analyse.

Une façon très simple de vérifier si un ensemble d'observations vérifie une loi repose sur le principe suivant : on coupe l'ensemble en deux, on estime les paramètres de la loi de BERNOULLI sur chacun des ensembles et ensuite on mesure la distance entre les deux estimations. De nombreuses autres approches sont possibles mais cette dernière nous permet une interprétation géométrique très simple.

Dans un premier temps, regardons ce qui se passe si nous calculons les fréquences sur une courbe discrète formée de quatre segments de droites (figure 2.44). Sur cet exemple, nous pouvons distinguer très nettement les changements de segments

discrets.

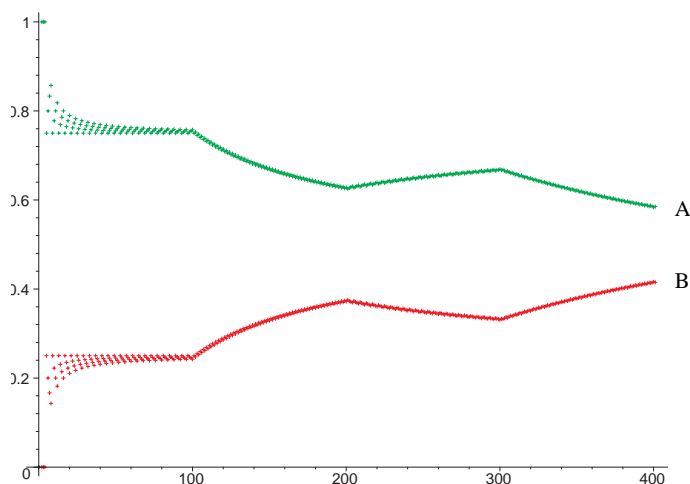


FIG. 2.44 —: Fréquences d'apparition des codes quand ceux-ci sont analysés suivant une abscisse croissante sur une courbe discrète formée de 4 segments de droite de 100 pixels et de pente $\frac{1}{4}$, $\frac{1}{2}$, $\frac{1}{4}$ et $\frac{2}{3}$.

Considérons le processus suivant : à partir d'un code M du codage de FREEMAN, nous allons maintenir deux fréquences \mathcal{G} et \mathcal{D} des codes respectivement à gauche et à droite de M . Ainsi, à chaque étape, nous mettons à jour \mathcal{G} avec l'ajout du premier code non visité à gauche de M . De la même façon, nous mettons à jour les fréquences de \mathcal{D} avec le premier code à droite de M non visité. Les ensembles \mathcal{G} et \mathcal{D} nous servent de test d'adéquation à la loi de BERNOULLI de l'ensemble des codes visités (de paramètres respectifs $p_{\mathcal{G}}$ et $p_{\mathcal{D}}$). Ainsi, nous pouvons nous intéresser à une distance entre les deux estimations afin de construire notre test statistique. D'un point de vue géométrique, l'ensemble \mathcal{G} (resp. \mathcal{D}) nous permet d'estimer les paramètres de la tangente à gauche en M (resp. à droite en M) ; la distance entre les mesures correspondrait à un test d'écart angulaire entre ces tangentes.

Dans ce qui suit, nous avons choisi la distance euclidienne entre les estimations mais de nombreuses autres métriques peuvent être utilisées. Ainsi, notre test d'adéquation est donné par la mesure d'erreur :

$$\epsilon = |p_{\mathcal{G}} - p_{\mathcal{D}}|$$

La figure 2.45 illustre cette mesure sur un exemple. Dans un premier temps, l'erreur ϵ décroît, ce qui nous permet d'avoir une bonne estimation des paramètres de la tangente discrète en M (voir paragraphe 5.3). Par la suite, une fois que les codes analysés sont issus d'autres loi de BERNOULLI, l'erreur croît. Le changement de loi est facilement détectable sur cet exemple puisque le long segment centré en M nous permet de bien estimer la tangente avant de changer de loi. Ainsi, si nous réduisons la taille du segment central (voir figure 2.46-(a) pour une taille de 25 pixels) le changement est toujours détectable mais moins précisément.

Bien évidemment, les lois associées aux segments adjacents à celui contenant M influent aussi sur la détection de la rupture. Ainsi, un cas pathologique pour cette méthode est illustré dans la figure 2.46-(b). Dans ce cas, les segments adjacents à M ont la même loi et donc si M est situé exactement au milieu du segment central, la dissymétrie entre les ensembles \mathcal{G} et \mathcal{D} n'est plus perceptible. Une idée simple qui pourrait résoudre ce problème consiste en l'utilisation d'un troisième ensemble \mathcal{A} qui serait un échantillonnage aléatoire des ensembles \mathcal{G} et \mathcal{D} .

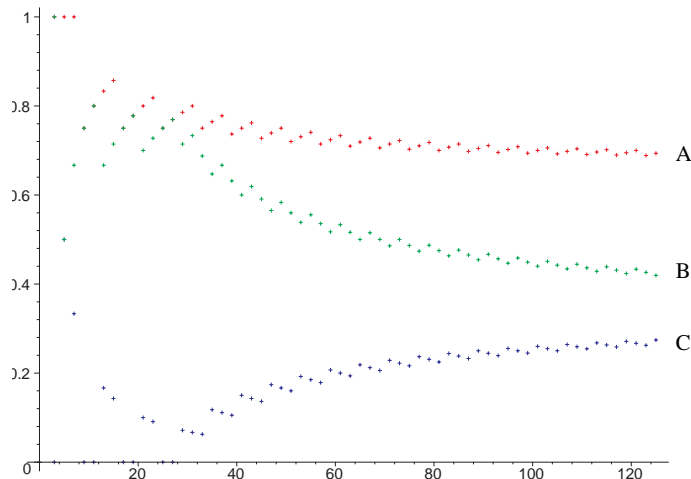
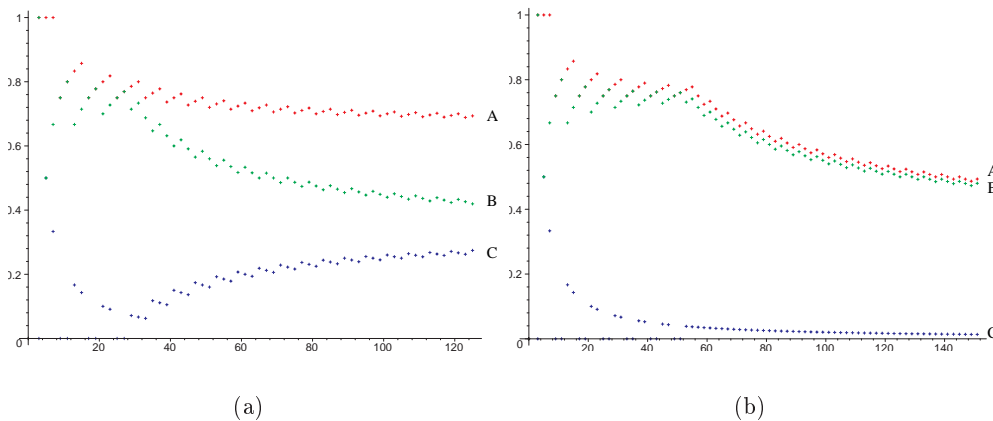
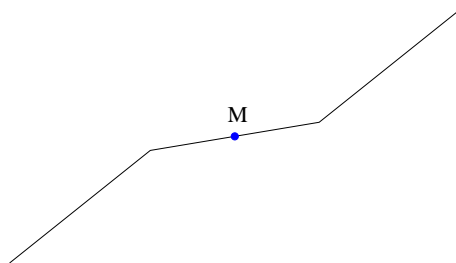


FIG. 2.45 –: Test d'adéquation à une loi de BERNOULLI pour la reconnaissance de droite : les courbes A et B indiquent les estimations p_G et p_D sur une courbe composée de 200 pixels, la courbe C correspond à la mesure d'adéquation ϵ . Les 50 premiers sont issus de la droite de pente $\frac{2}{3}$, les 100 suivants de la droite de pente $\frac{1}{4}$ et les 50 derniers de la droite de pente $\frac{1}{3}$. Le point M correspond au pixel d'abscisse 105.



(a)

(b)



(c)

FIG. 2.46 –: Illustration du test statistique : (a) exemple où le segment central est plus petit (25 pixels avec les mêmes paramètres) que dans la figure 2.45, (b) cas pathologique où les segments à l'extrémité ont la même loi et où M est au milieu du segment central ; (c) illustration géométrique de ce cas pathologique.

Dans le cas tridimensionnel, nous avons le même type de propriété : soit un plan discret naïf $\mathcal{P}(a, b, c, \mu)$ dans le premier 48^{ème} d'espace (*i.e.* $0 \leq a \leq b < c$). Étant donnée la classification des surfels de la figure 2.47, nous avons :

Proposition 2.10 *Considérons le processus qui consiste à choisir aléatoirement, sur ce plan infini, un surfel. La probabilité d'obtenir un surfel de type 3 est de $\frac{b}{c}$, la probabilité d'obtenir un surfel de type 2 est de $\frac{a}{c}$ et la probabilité d'obtenir un surfel de type 1 est de $1 - \frac{b}{c} - \frac{a}{c}$*

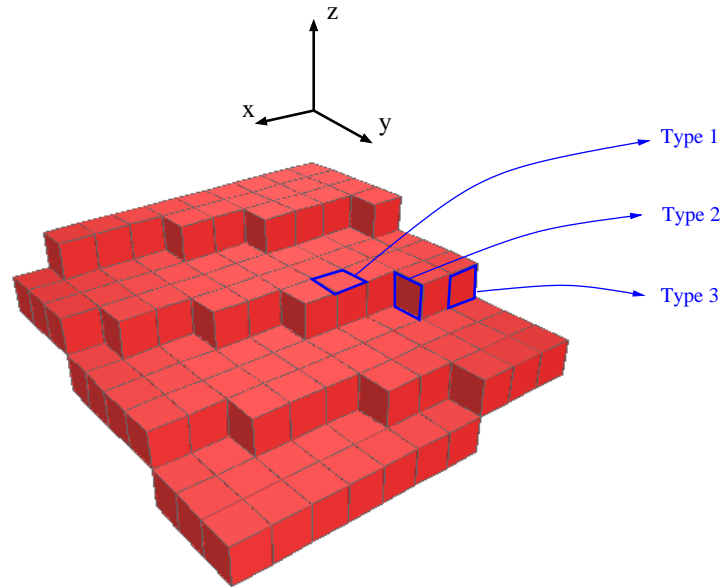


FIG. 2.47 –: Morceau de plan discret $\mathcal{P}(7, 17, 57, 0)$ et classification des différents surfels.

La preuve de cette proposition est directe par décomposition du plan discret en droites discrètes.

Dans ce cas, on ne parle plus de loi de BERNOULLI mais de loi empirique de paramètres $(p_1 = \frac{b}{c}, p_2 = \frac{a}{c})$. De la même manière que pour le cas 2D, un test d'adéquation à une loi empirique des types de surfels correspond à une reconnaissance statistique de plan discret. Étant donné la complexité et le coût des algorithmes de reconnaissance classiques de plan discret, l'enjeu pour le cas 3D est important. Par exemple, si nous voulons estimer la normale en un point d'une surface discrète, une approche consisterait en le calcul du plan discret tangent, c'est-à-dire du plan discret le plus grand qu'il est possible de reconnaître centré au point considéré. Dans ce cas, si nous voulons la normale en tout surfel de la surface, l'utilisation des algorithmes classiques de reconnaissance entraîne un coût prohibitif. Pour cela, avoir un estimateur statistique de plan tangent serait très intéressant.

De même qu'en 2D, nous pouvons imaginer un test qui consiste à découper les observations en deux ensembles, à estimer les lois séparément et ensuite à évaluer l'adéquation en comparant les résultats. Pour cela, la figure 2.48 présente deux approches possibles.

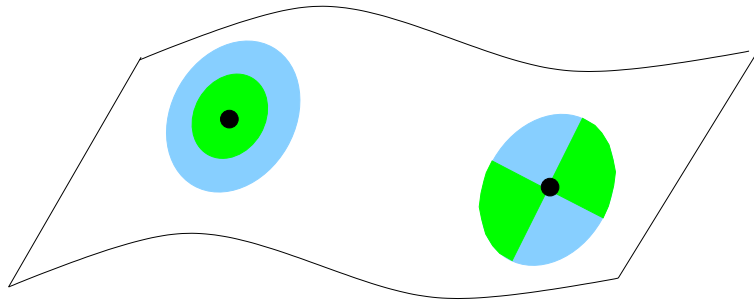


FIG. 2.48 –: *Quelques pistes pour la création d'un test d'adéquation à la loi empirique : la première se base sur une décomposition en deux disques géodésiques de taille croissante, la seconde sur une décomposition en quadrant surfacique.*

2.6 Conclusion

Dans ce chapitre, nous avons présenté les différentes approches pour la synthèse et la reconnaissance d'objets géométriques simples que sont les droites et plans discrets. Non seulement l'étude de ces objets présente un intérêt théorique très important, mais en plus, nous verrons par la suite qu'ils sont essentiels pour l'analyse de forme ou l'estimation de nombreuses mesures euclidiennes comme la longueur, l'aire ou encore les normales ou l'estimation de courbure.

Dans le dernier paragraphe, nous avons présenté une approche nouvelle du problème de reconnaissance de droites et de plans discrets basée sur une analyse statistique de ces objets. Cette alternative statistique à la géométrie nécessite maintenant d'être utilisée dans des applications réelles.

Chapitre 3

Le cercle discret

Sommaire

3.1	Introduction	80
3.2	Synthèse et analyse du cercle discret	80
3.3	Reconnaissance et segmentation	83
3.3.1	État de l'art	83
3.3.2	Approche basée sur l'analyse des paraboles discrètes	84
3.3.3	Le problème de séparation par arc	86
3.3.4	Quelques optimisations au calcul de l' <i>acd</i>	87
3.3.5	Approche arithmétique pour la reconnaissance de cercle discret	89
3.3.5.1	Ligne polygonale quasi-circulaire dans \mathbb{Z}^2	89
3.3.5.2	Cercle discret et quasi-circularité	93
3.3.5.3	Algorithmes de reconnaissance et segmentation	93
3.3.5.4	Analyse de complexité	97
3.3.6	Quelques résultats	98
3.3.7	Comparaison avec KOVALEVSKY (1990)	99
3.4	Conclusion	101

3.1 Introduction

Dans ce chapitre, nous nous intéressons au cercle discret. L'objectif est non seulement d'analyser et de reconnaître un nouvel objet mais aussi d'offrir la possibilité d'estimer la courbure en un point d'une courbe discrète par l'utilisation de cercles osculateurs comme nous le verrons dans le chapitre 5.

Nous commençons notre analyse par les algorithmes classiques de tracé de cercles sur la grille discrète. Ensuite, nous nous intéressons aux différents algorithmes de reconnaissance de cercles discrets et proposons un algorithme incrémental et simple à implémenter pour la segmentation d'une courbe discrète en arcs de cercle (COEUR-JOLLY ET AL., 2001c).

3.2 Synthèse et analyse du cercle discret

Comme pour la droite, une approche naturelle pour définir un cercle discret consiste à superposer un cercle réel sur une grille discrète et ensuite à choisir, en utilisant un processus de discrétisation donné, l'ensemble des pixels *concernés* par ce cercle. Le choix du processus de discrétisation se fait de façon à approximer *au mieux* le cercle euclidien.

Par exemple, NAKAMURA et AIZAWA (1984) considèrent une discrétisation *au plus proche* de type GIQ (figure 3.1-(b)) ou KIM (1984a) une discrétisation OBQ du cercle euclidien (figure 3.1-(c)).

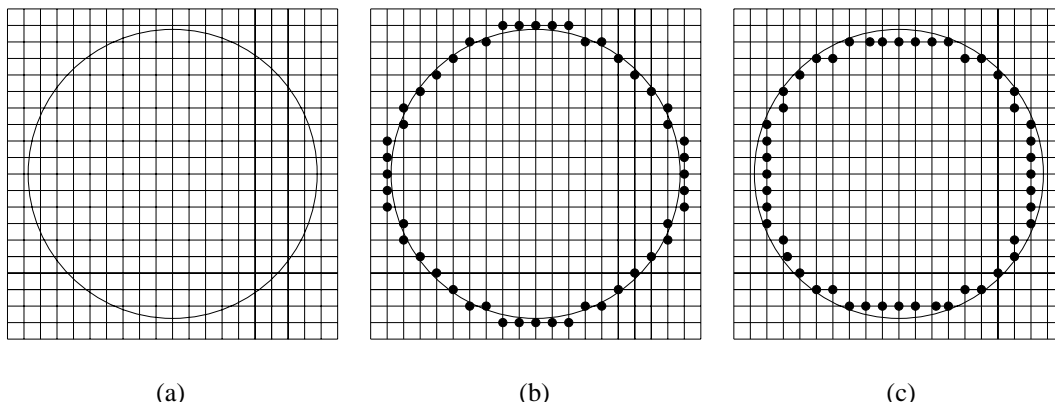


FIG. 3.1 — Discretisation de cercles euclidiens : (a) cercle euclidien superposé sur une grille discrète, (b) discretisation GIQ de NAKAMURA et AIZAWA et (c) discretisation de KIM.

Dans une approche similaire au tracé de droite, BRESENHAM (1977) a proposé une définition de cercle discret et un algorithme de tracé très simple. Celui-ci s'effectue dans le second octant (*i.e.* $y > x \geq 0$), les autres octants étant obtenus par symétrie, et commence par le pixel $(0, R)$ pour un cercle de rayon R . En traçant le cercle en x croissant, nous avons à chaque étape le choix entre deux pixels : le pixel $(x + 1, y)$ ou le pixel $(x + 1, y - 1)$. Ce choix se fait en calculant la distance minimale au cercle euclidien (voir algorithme 11).

Tous les cercles discrets présentés ci-dessus correspondent à des approximations du cercle réel. Cependant, la famille des cercles concentriques à partir du point $(0, 0)$ peut présenter des trous ou des chevauchements. Ces propriétés peuvent s'avérer essentielles pour certaines applications (TOUGNE, 1997).

Algorithme 11 Tracé de cercle de BRESENHAM (1977)

```

TRACÉ_BRESENHAM( $R$ )
1:  $x = 0, y = r$ 
2:  $d = 5/4 - R$ 
3: AFFICHE_PIXEL( $x, y$ )
4: tant que  $y > x$  faire
5:   si  $d < 0$  alors
6:      $d += 2x + 3$ 
7:   sinon
8:      $d += 2x - 2y + 5$ 
9:      $y -= 1$ 
10:  fin du si
11:  AFFICHE_PIXEL( $x, y$ )
12:   $x += 1$ 
13: fin du tant que

```

Une classe de cercles vérifiant ces propriétés a cependant été proposée par ANDRÈS (1994); ANDRÈS (1994). Ces cercles sont appelés *cercles arithmétiques* et sont définis par l'ensemble des points (x, y) de la grille vérifiant (voir figure 3.2) :

$$\left(R - \frac{1}{2}\right)^2 \leq (x - x_0)^2 + (y - y_0)^2 < \left(R + \frac{1}{2}\right)^2$$

avec $R, x_0, y_0 \in \mathbb{Z}$.

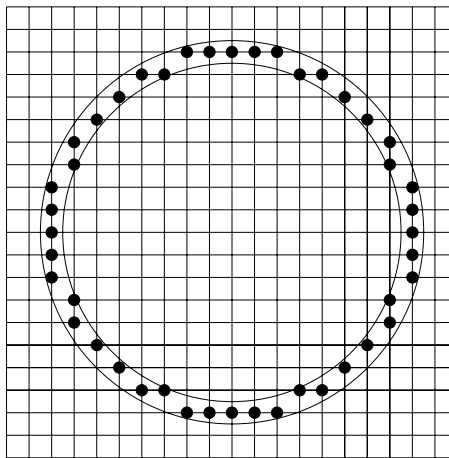


FIG. 3.2 –: Cercle arithmétique de rayon 8.

Par la suite, ANDRÈS et JACOB (1997) ont proposé une extension de cette définition pour les hypersphères analytiques discrètes. Ces hypersphères pavent l'espace et se caractérisent par des coordonnées des centres, des rayons et des épaisseurs non entiers.

Nous terminons notre introduction par le cercle de PITTEWAY (1974) ainsi que des cercles dérivés. Considérons les intersections d'un cercle réel de rayon R avec la grille discrète donnée par le système :

$$\begin{cases} x^2 + y^2 = R^2 \\ x = R - k \\ -R \leq k \leq R \end{cases}$$

Dans le premier octant, nous obtenons le système suivant :

$$\begin{cases} y = \sqrt{2kx + k^2} \\ x = R - k \\ 1 \leq k \leq R \end{cases}$$

Nous définissons le champ de paraboles discrètes \mathcal{H}_k comme étant une discrétisation du champ de paraboles réelles $y = \sqrt{2kx + k^2}$. Ainsi, si nous considérons les $y = \lceil \sqrt{2kx + k^2} \rceil$, nous définissons le cercle PITTEWAY (1974). De la même manière, nous pouvons définir les cercles *planchers* et *plafonds* en considérant respectivement les paraboles $y = \lfloor \sqrt{2kx + k^2} \rfloor$ et $y = \lceil \sqrt{2kx + k^2} \rceil$ (voir TOUGNE 1997).

Le tracé de tels cercles se fait dans le premier octant et consiste en une succession de pans verticaux guidés par les paraboles discrètes (voir figure 3.3 et algorithme 12).

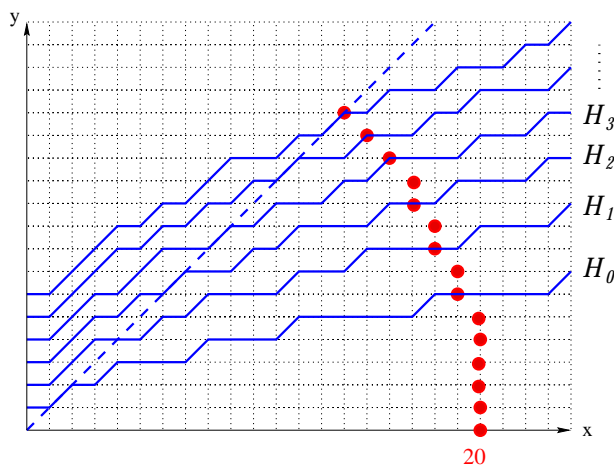


FIG. 3.3 – Exemple de construction d'un cercle de Pitteway plancher de rayon 20.

Algorithme 12 Tracé de cercle de PITTEWAY (1974)

```

TRACÉ_PITTEWAY(R)
1:  $x = r, y = 0$ 
2:  $k = 1$ 
3: tant que  $y \leq x$  faire
4:   tant que  $y < \lceil \sqrt{2kx + k^2} \rceil - 1$  faire
5:     AFFICHER_PIXEL( $x, y$ )
6:      $y += 1$ 
7:   fin du tant que
8:   AFFICHER_PIXEL( $x, y$ )
9:   si  $y = \lceil \sqrt{2kx + k^2} \rceil$  alors
10:    AFFICHER_PIXEL( $x - 1, y$ )
11:  fin du si
12:   $k += 1$ 
13:   $x -= 1$ 
14:   $y += 1$ 
15: fin du tant que
    
```

En se basant sur cette analyse des paraboles discrètes, TOUGNE (1997) montre qu'il est possible de générer de tels cercles sur automates cellulaires et que les cercles

de BRESENHAM et les cercles arithmétiques peuvent se construire sur le même principe. Plus précisément, le faisceau des paraboles associé aux cercles arithmétiques est donné par :

$$\begin{cases} y = \left[\sqrt{x(1+2k) - k^2 + 1/4} \right] \\ x = R - k \\ 0 \leq k \leq R \end{cases}$$

3.3 Reconnaissance et segmentation

3.3.1 État de l'art

Dès les premières applications de reconnaissance de formes, les estimations de circularité ou de compacité se sont avérées être des mesures importantes. Du point de vue de la géométrie discrète, les premiers travaux sur la reconnaissance de cercles ou de disques discrets sont apparus dès le début des années 80.

D'une manière générale, on dira qu'un ensemble de pixels appartient à un cercle discret s'il existe, étant donné un processus de discrétisation, un cercle réel dont la discrétisation contient l'ensemble des pixels.

Comme pour les droites, deux problèmes distincts existent :

- le premier consiste en un test de circularité, nous testons juste l'existence d'un cercle réel solution ;
- le second consiste en une reconnaissance du cercle, c'est-à-dire au calcul de l'ensemble des cercles réels solutions. Nous définissons ainsi ce que l'on appellera *la pré-image* de l'ensemble de pixels.

Dans un premier temps, NAKAMURA et AIZAWA (1984) ont proposé un algorithme de test de circularité basé sur leur définition de cercle discret. Les algorithmes proposés ont cependant une complexité exponentielle au pire cas (les auteurs donnent quelques pistes pour rendre les algorithmes polynomiaux dans la conclusion).

La même année, KIM (1984a) a proposé un algorithme permettant de décider si un ensemble de pixels contenus dans une image $N \times N$ est un disque discret. Cet algorithme initial a une complexité en $O(N^3)$ mais KIM et ANDERSON (1984) optimisent ce calcul pour obtenir une complexité en $O(N^2)$.

Par la suite, FISK (1986) propose une reconnaissance de cercles discrets basée sur le principe de séparation de points par des cercles sur lequel nous reviendrons ultérieurement. Cet algorithme a une complexité en $O(n^2)$ où cette fois n représente le nombre de pixels de la courbe discrète.

Pour le problème du test de circularité, SAUER (1993) propose un algorithme très complexe dont le coût est linéaire (*i.e.* $O(n)$) en le nombre de pixels de la courbe discrète. Cet algorithme se ramène à un problème de couverture par un cercle minimal dont la complexité linéaire a été montrée par MEGIDDO (1983). Cette approche n'est cependant pas adaptée au cas de morceaux de cercle.

DAMASCHKE (1995) montre une réécriture du problème du test de circularité en un système d'inéquations linéaires de dimension 3. Ainsi, en utilisant le théorème de MEGIDDO (1984) (voir théorème 2.2), il obtient un algorithme en temps linéaire.

WORRING et SMEULDERS (1995) proposent une estimation de circularité basée sur une construction de la pré-image de courbes de taille fixe. Plus précisément, ils

proposent un calcul de l'ensemble des cercles réels solutions pour des morceaux de courbes de taille r , puis présentent un estimateur statistique permettant d'évaluer le rayon d'un cercle solution pour une courbe complète. La qualité de cette approximation dépend bien évidemment du paramètre r .

Enfin, KOVALEVSKY (1990) présente un algorithme de reconnaissance basé sur la séparation de deux ensembles de points par des cercles. Nous reviendrons plus en détail sur ce travail dans le paragraphe suivant. La complexité de cet algorithme n'est pas donnée par l'auteur mais comme cette approche se ramène à un calcul d'intersection de n^2 inéquations en dimension 2, nous évaluons la complexité à $O(n^2 \log(n))$ en utilisant l'algorithme de PREPARATA et SHAMOS (1985) (voir paragraphe 3.3.7).

Le tableau 3.3.1 résume les différentes approches présentées.

Algorithme	Complexité en temps	Incrémental	Problème
NAKAMURA et AIZAWA (1984)	-	non	circularité
KIM (1984a)	$O(N^3)$ (image $N \times N$)	non	circularité
KIM et ANDERSON (1984)	$O(N^2)$	non	circularité
FISK (1986)	$O(n^2)$	non	reconnaissance
SAUER (1993)	$O(n)$	non	circularité
DAMASCHKE (1995)	$O(n)$	oui avec BUZER (2002)	circularité
WORRING et SMEULDERS (1995)	-	-	estimateur, circularité
KOVALEVSKY (1990)	$O(n^2 \log(n))$	oui	reconnaissance

TAB. 3.1 –: Algorithmes et complexités pour la reconnaissance de cercles discrets.

3.3.2 Approche basée sur l'analyse des paraboles discrètes

Durant mon stage de DEA (COEURJOLLY, 2000), nous nous sommes intéressés à une représentation des cercles discrets par un codage lié aux paraboles discrètes. Nous avons aussi essayé de proposer un algorithme de reconnaissance sur ce codage mais sans succès. Nous présentons ici le codage proposé dont nous allons nous servir par la suite pour un estimateur de courbe discrète très rapide (voir paragraphe 5.5.2).

Nous considérons dans un premier temps une courbe discrète $\{(x_i, y_i)\}_{i=1..n}$ telle que $x_{i+1} \leq x_i$ et $y_{i+1} \geq y_i$ pour tout i .

Nous définissons sur cette courbe la notion point de rupture :

Définition 3.1 (point de rupture) *Un point (x_i, y_i) est appelé point de rupture si et seulement si le point (x_{i-1}, y_{i-1}) est tel que $x_i \neq x_{i-1}$.*

Il est clair que les points de rupture d'un arc de cercle discret de rayon R dans le premier octant correspondent exactement aux intersections entre le champ de paraboles discrètes \mathcal{H}_k et les droites verticales $x = R - k$.

Nous définissons également un code de rupture d'une courbe discrète :

Définition 3.2 (code de rupture) *Le code de rupture B_c d'une courbe discrète est défini comme étant la hauteur en nombre de points entre deux points de rupture consécutifs.*

Un exemple de points de rupture et de code de rupture est présenté en figure 3.4.

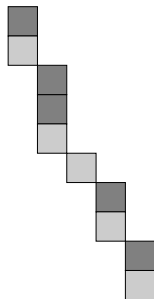


FIG. 3.4 – : *Un exemple de courbe discrète (les points de rupture sont représentés en gris clair) et le code de rupture de ce segment est : 2-2-1-3-2.*

Nous proposons de construire un algorithme de reconnaissance de cercle basé sur le code de rupture. Pour cela, nous allons fixer le type de cercle discret à reconnaître en fixant le champ de paraboles discrètes. Nous considérons donc les paraboles :

$$y = \left[\sqrt{R(1 + 2k) - k^2 + 1/4} \right]$$

avec $0 \leq k \leq R$

qui correspondent aux cercles arithmétiques. Cette formule s'obtient facilement en calculant les intersections entre la grille discrète et le cercle $x^2 + y^2 = (R + 1/2)^2$.

Nous pouvons maintenant lier les codes de rupture et les cercles arithmétiques :

Proposition 3.1 *Soit B_c un code de rupture d'un arc de cercle discret arithmétique de rayon R dans le premier octant, nous avons alors :*

$$\sum_{i=0}^k B_c(i) = \left[\sqrt{R(1 + 2k) - k^2 + 1/4} \right]$$

En se basant sur cette proposition, nous pouvons inverser l'équation précédente pour obtenir le rayon du cercle discret, ou plutôt l'intervalle des rayons possibles, en fonction d'un code de rupture :

Proposition 3.2 *Soit B_c un code de rupture d'un arc de cercle arithmétique de rayon R dans le premier octant, notons h la hauteur de la $k^{\text{ième}}$ parabole discrète (hauteur du $k^{\text{ième}}$ point de rupture, $h = \sum_{i=0}^k B_c(i)$), l'ensemble \mathcal{R}_h des rayons des cercles discrets possibles ayant B_c comme code de rupture jusqu'au $k^{\text{ième}}$ élément, est donné par la formule :*

$$\mathcal{R}_h = \left\{ \left[\frac{(h - 1/2)^2 + k^2 - 1/4}{1 + 2k} \right], \dots, \left[\frac{(h + 1/2)^2 + k^2 - 1/4}{1 + 2k} \right] \right\}$$

Par exemple, dans le cas du cercle de Pitteway plancher de la figure 3.3, si l'on considère la première parabole \mathcal{H}_0 , l'ensemble des rayons possibles pour une hauteur

$h = 5$ est [18...23]. Cette incertitude sur le rayon correspond aux paliers des paraboles discrètes.

Ainsi, si la courbe discrète correspond à un arc de cercle arithmétique et si nous sommes sûrs que le premier élément dans le code de rupture correspond au passage de la première parabole, nous pouvons alors reconnaître le cercle discret en calculant, par l'équation précédente, l'ensemble des rayons possibles. Cet ensemble se réduit en un seul élément si la courbe est exactement le huitième du cercle arithmétique (nous pouvons alors reconstruire tout le cercle par symétrie de cette courbe).

Pour obtenir un algorithme de reconnaissance de cercles arithmétiques à partir d'une courbe discrète quelconque, l'idée est de fixer le champ de paraboles en fonction du code de rupture, et ensuite à partir du champ de paraboles retrouver le ou les rayon(s) de cercle(s) arithmétique(s) possible(s). En fait, le problème peut se résumer de la façon suivante : chaque élément du code de rupture est une fonction de deux paramètres, R le rayon du cercle à retrouver et k la parabole discrète à laquelle il correspond. Nous avons vu avec la proposition précédente que le calcul de ces paramètres nécessite la connaissance d'un morceau de code de rupture qui correspondrait à un huitième de cercle. Dans le cas général, nous avons un morceau de code de rupture dont on ne sait pas à partir de quel indice de parabole il débute. Pour pouvoir proposer un algorithme de reconnaissance, il nous faudrait :

- Définir la notion de code de rupture dans tous les octants. Il faut pour cela détecter dans quel octant est susceptible de se trouver l'arc de cercle et ensuite calculer le code de rupture approprié.
- Trouver dans ce code de rupture toutes les sous-suites monotones. En effet, il est clair que dans le cas d'un arc de cercle dans le premier octant, le code de rupture est décroissant. Nous réduisons, par ce processus, l'ensemble des sous-suites du code.
- Parmi ces sous-suites, trouver la plus longue en tout point qui vérifie la proposition 3.2, c'est-à-dire trouver les indices des paraboles auxquels la sous-suite correspond.
- Extraire en tout point un rayon et un centre du cercle arithmétique estimés.

Cet algorithme reste encore très compliqué à mettre en œuvre et coûteux en terme de complexité. Nous verrons par la suite (paragraphe 5.5.2) l'utilité d'une telle analyse pour l'estimation de courbure en un point d'une courbe discrète.

3.3.3 Le problème de séparation par arc

En géométrie algorithmique, la séparation de deux ensembles de points par des cercles est un problème classique. Nous verrons par la suite l'intérêt de cet algorithme pour le cas des cercles discrets.

Définition 3.3 (Séparabilité par un cercle) Soient S et T deux ensembles finis de points dans le plan, S est séparable de T par un cercle s'il existe un cercle réel de centre ω et de rayon R tel que :

$$\forall s \in S, \forall t \in T \quad s \in D(\omega, R) \text{ et } t \notin D(\omega, R)$$

où $D(\omega, R)$ est le disque centré en ω et de rayon R .

Dans la définition précédente, nous avons considéré sans perte de généralité que l'ensemble S est intérieur au disque.

Nous reprenons les notations de COEURJOLLY ET AL. (2001c) et définissons le polygone des solutions :

Définition 3.4 (Domaine des centres des arcs) *Soient S et T deux ensembles finis de points tels que S est séparable de T par un cercle. Le domaine des centres des arcs, noté $acd(S, T)$ (pour arc center domain), est le lieu des centres ω des cercles séparant S de T .*

Il est clair qu'à partir de ce domaine et de S et T , il est possible de retrouver les rayons R des cercles séparant.

Si un cercle sépare S et T , son centre ω vérifie nécessairement l'inégalité :

$$\forall s \in S, \forall t \in T \quad dist(\omega, s) < dist(\omega, t)$$

où $dist(a, b)$ est la distance euclidienne usuelle.

Ainsi, le domaine des centres, ou $acd(S, T)$, est un polygone convexe dans le plan (pouvant être vide) qui correspond à l'intersection d'inéquations linéaires définies par des médiatrices entre les points de S et les points de T :

Lemme 3.1 *Soient $s \in S$ et $t \in T$ et $H(s, t)$ l'inéquation linéaire définie par la médiatrice du segment $[st]$ et telle qu'elle contient s . Nous avons alors :*

$$acd(S, T) = \bigcap_{s \in S, t \in T} H(s, t)$$

Un tel polygone convexe est aussi appelé *cellule de Voronoï généralisée* (voir par exemple PREPARATA et SHAMOS 1985 ou DE BERG ET AL. 2000). La figure 3.5 montre un exemple d' acd entre deux ensembles de points.

Le problème de séparation par arc se ramène donc au calcul de l'espace des solutions dans un problème de programmation linéaire en dimension 2. Ainsi, nous avons choisi, dans ce qui suit, d'implémenter ce calcul d' acd par l'algorithme de PREPARATA et SHAMOS (1985) basé sur une transformation dans l'espace dual (voir annexe C). Nous rappelons les caractéristiques et avantages de cet algorithme :

- calcul de l'espace des solutions de n contraintes linéaires en dimension 2 en $O(n \log(n))$ ce qui est optimal pour le problème ;
- implémentation très simple basée sur des outils classiques de géométrie algorithmique (dualité point/droite et enveloppe convexe) ;
- mise à jour incrémentale du domaine en $O(\log(n))$.

En utilisant donc cet algorithme, le calcul du domaine des centres des cercles séparant S de T a une complexité en $O(|S||T| \log(|S||T|))$.

3.3.4 Quelques optimisations au calcul de l' acd

Dans COEURJOLLY ET AL. (2001c), certaines observations ont été faites concernant des optimisations au calcul de l' acd . Ces réductions ne permettent pas d'améliorer la complexité au pire cas de l'algorithme mais permettent d'en augmenter l'efficacité.

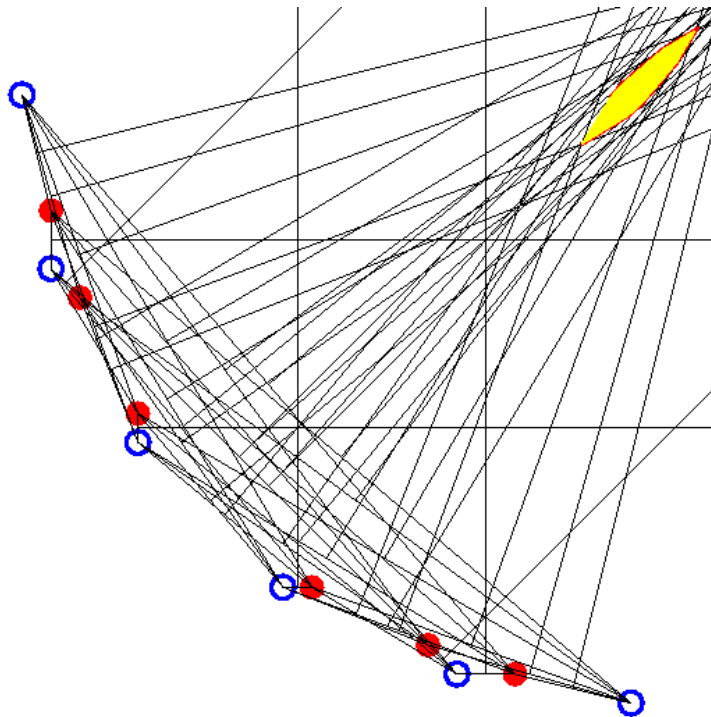


FIG. 3.5 –: Exemple de domaine des centres des cercles séparant deux ensembles de points du plan construit par intersection de contraintes issues des médiatrices.

Réduction de S

Si un cercle sépare S de T , celui-ci contient nécessairement l'enveloppe convexe de S . Ainsi, si nous remplaçons S par l'ensemble des sommets de son enveloppe convexe, cela ne change pas la forme de l'*acd*.

Cette optimisation a un coût en $O(|S|\log(|S|))$. Étant donné que nous n'avons aucune hypothèse sur S , nous ne pouvons quantifier le gain de cette réduction puisque dans le pire cas (dans le cas où tous les points de S sont sommets de l'enveloppe convexe de S), aucun point n'est enlevé.

Réduction de T

Considérons deux points t et t' de T et supposons que l'enveloppe convexe des points $S \cup \{t\}$ contient t' . Alors, s'il existe un cercle contenant S mais qui ne contient pas t' , ce cercle ne contient pas t . En d'autres termes, le point t peut être supprimé de T (voir figure 3.6-(a)).

Nous obtenons donc le processus suivant : pour tous les points t de T , on construit l'enveloppe convexe de $S \cup \{t\}$. Si cette enveloppe convexe contient un autre point t' de T , le point t est supprimé.

Finalement, nous obtenons un algorithme de réduction de T en $O(|T|\log(|T|))$ mais à nouveau, nous ne pouvons pas quantifier le gain dans le cas général.

En combinant ces deux réductions, nous obtenons la simplification illustrée dans la figure 3.6-(b).

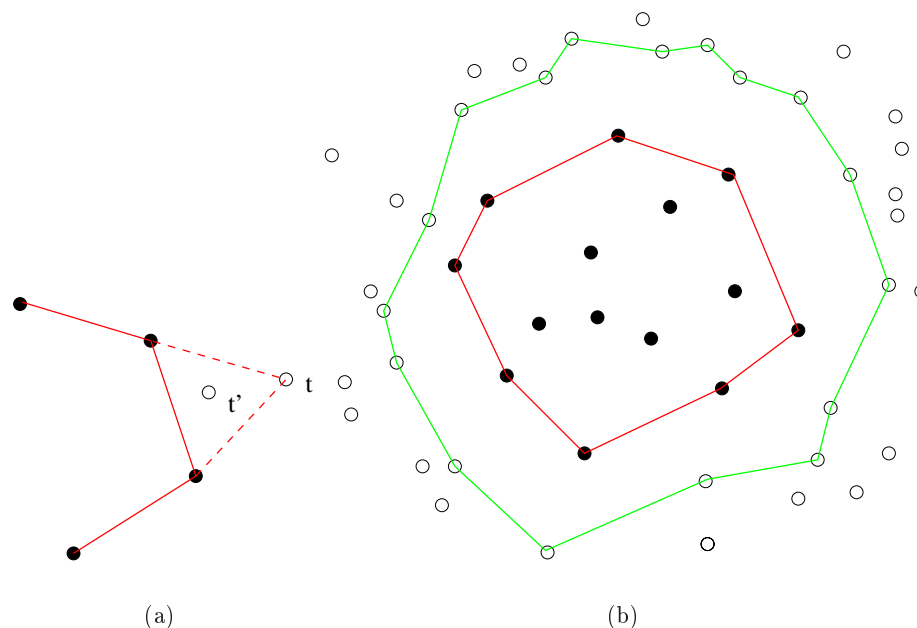


FIG. 3.6 – : *Optimisation du calcul du cercle séparateur par réduction des ensembles S et T : les points de S sont en noir et les points de T en blanc, (a) illustration de la réduction de T , dans ce cas, t est supprimé et (b) processus global avec les deux réductions, les points restants sont uniquement les sommets des deux polygones.*

3.3.5 Approche arithmétique pour la reconnaissance de cercle discret

Dans cette partie, nous considérons des domaines S et T adaptés à notre problème de reconnaissance de cercle discret. Dans un premier temps, nous nous intéressons aux courbes polygonales quasi-circulaires (voir définition ci-dessous) sur la grille discrète et à leurs propriétés arithmétiques. Puis nous verrons les liens entre ces courbes et les cercles discrets.

3.3.5.1 Ligne polygonale quasi-circulaire dans \mathbb{Z}^2

Pour simplifier les explications et sans perte de généralités, nous considérons deux points discrets notés p et q tels que le segment $[pq]$ soit dans le premier octant. De plus nous supposons qu'il existe deux entiers a, b (avec $\text{pgcd}(a, b) = 1$), tels que :

$$\vec{pq} = (a, b)^T$$

On notera $\vec{u} = (a, b)^T$.

Nous supposons aussi connu un vecteur de BEZOUT, noté \vec{v} , associé à $(a, b)^T$ (voir paragraphe 2.2.1.2 pour plus de détails sur ce vecteur) tel que :

$$\det(\vec{u}, \vec{v}) = 1$$

Avant de continuer, nous définissons une famille de points discrets très importants pour notre problème :

Définition 3.5 (Points de BEZOUT) *On appelle points de BEZOUT associés aux vecteurs \vec{u} et \vec{v} , l'ensemble des points discrets \mathcal{B} tels que :*

$$\vec{p\mathcal{B}} = \vec{v} + k\vec{u}$$

avec $k \in \mathbb{Z}$.

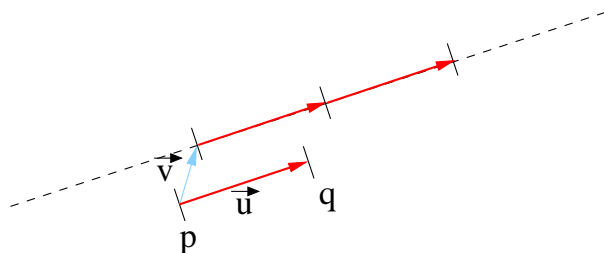


FIG. 3.7 – : Illustration des différentes définitions : un segment discret de vecteur directeur \vec{u} dans le premier octant, le vecteur de BEZOUT \vec{v} associé et les points de BEZOUT qui se trouvent sur la droite en pointillé et qui sont générés par \vec{u} .

Cette notion de points de BEZOUT d'un segment $[pq]$ est proche de la notion de *points faiblement extérieurs* dans le cas des droites discrètes (DEBLED-RENNESON, 1995). La figure 3.7 illustre les différentes définitions.

Considérons maintenant le problème de séparation par un cercle des pixels au dessus de $[pq]$, formant l'ensemble T , de ceux sous ce segment (ensemble S). Cet exemple est illustré figure 3.8-(a). Dans ce cas, nous pouvons réduire l'ensemble des points S aux points p et q pour les raisons exposées dans le paragraphe 3.3.4. De même, nous avons le théorème suivant :

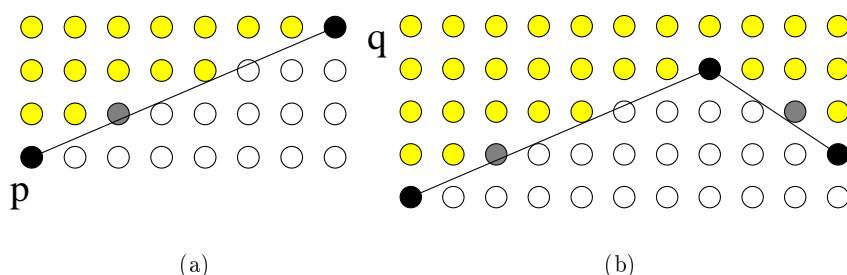


FIG. 3.8 – : (a) séparation du segment $[pq]$ de tous les pixels au dessus grâce au point de BEZOUT (en gris) et (b) points de BEZOUT associés à une courbe polygonale convexe.

Théorème 3.1 *Considérons le triangle formé de p , q et d'un point t de T , les seuls points t tels que le triangle (p, q, t) (arêtes incluses) ne contienne aucun point discret sont les points de BEZOUT de $[pq]$.*

Preuve : Dans un premier temps, nous montrons que le triangle associé à un point de BEZOUT ne contient aucun point discret. Ce point est trivial puisque que \vec{u} et \vec{v} sont uni-modulaires (le déterminant est nul) et donc il n'y a aucun point discret dans le parallélogramme formé de ces vecteurs (voir par exemple HARDY et WRIGHT 1975). Ainsi, il ne peut y avoir de points dans le triangle (p, q, t) qui correspond au parallélogramme ci-dessus coupé par une diagonale. Ensuite, supposons un point t qui ne soit pas un point de BEZOUT, donné par :

$$\vec{pt} = k\vec{u} + k'\vec{v}$$

avec $k, k' \in \mathbb{Z}$ et $k' > 1$.

Nous utilisons le théorème de PICK classique en théorie des nombres (HARDY et WRIGHT, 1975) qui lie le nombre de points I intérieurs à un polygone discret au nombre de points B sur le bord du polygone et à l'aire A de celui-ci par la formule :

$$I = A - \frac{B}{2} + 1$$

Dans notre cas, l'aire du triangle est donnée par le déterminant :

$$A = \frac{|\det(\vec{pt}, \vec{pq})|}{2}$$

Et donc :

$$\begin{aligned} 2I &= |\det(\vec{pt}, \vec{pq})| - B + 2 \\ &= |k||k'| |\det(\vec{u}, \vec{v})| - B + 2 \\ &= |k||k'| - B + 2 \end{aligned}$$

Si un point discret t' est sur une arête du triangle (p, q, t) , nous ne satisfaisons pas l'hypothèse du théorème qui inclus les arêtes des bords dans l'énumération des points intérieurs. Nous pouvons donc affecter à B la valeur 3 :

$$2I \geq |k||k'| - 1$$

Or k est nécessairement différent de 0, sinon nous aurions au moins un point t' sur l'arête $[pt]$. Si maintenant $|k'| > 2$, le problème est résolu puisque nous avons :

$$2I \geq 3|k| - 1$$

avec $|k| > 0$.

Considérons maintenant le cas $k' = 2$. Si $k = 1$, nous avons alors un point t' sur $[qt]$ et enfin, pour $k = -1$, nous avons aussi un point sur $[qt]$. Ceci implique finalement que $|k| > 1$ et dans ce cas :

$$2I \geq 3$$

ce qui prouve le théorème puisque nous avons au moins un point t' dans (p, q, t) .
□

Nous avons donc le corollaire suivant :

Corollaire 3.1 *Étant donné un segment $[pq]$ dans le premier octant, l'ensemble des centres des cercles, séparant ce segment de tous les points qui sont au-dessus de celui-ci, est donné uniquement par :*

$$\text{acd}(\{p, q\}, \{\mathcal{B}\})$$

où \mathcal{B} est le point de BEZOUT le plus proche de la médiatrice de $[pq]$.

En utilisant le principe de réduction de T du paragraphe 3.3.4, la preuve de ce corollaire est directement induite du théorème ci-dessus. Le point de BEZOUT \mathcal{B} est choisi en fonction de sa distance à la médiatrice de $[pq]$ puisque c'est le point le plus contraignant. En effet, si un cercle sépare p et q de \mathcal{B} , il sépare tous les autres points de BEZOUT de $[pq]$.

En d'autres termes, pour résoudre le problème de séparation de $[pq]$ de tous les pixels au dessus par un arc de cercle, il suffit de considérer uniquement deux contraintes linéaires générées par les points p , q et \mathcal{B} (voir figure 3.8-(a)).

Dans le cas où, $[pq] = k(a, b)^T$ avec $k \in \mathbb{Z}$, $k > 1$ et $\text{pgcd}(a, b) = 1$, il nous suffit de considérer le point de BEZOUT le plus proche de la médiatrice (voir figure 3.9). Dans ce cas, le théorème précédent peut être prouvé en utilisant exactement les mêmes techniques.

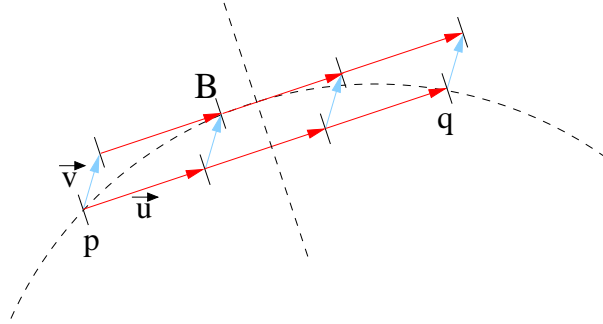


FIG. 3.9 —: *Illustration du choix du point de BEZOUT quand la pente de $[pq]$ n'est pas une fraction irréductible.*

Remarquons que l'hypothèse selon laquelle $[pq]$ doit être dans le premier octant peut être maintenant relâchée. En effet, quelle que soit l'orientation de $[pq]$, on peut associer deux points de BEZOUT à ce segment, le premier permettant d'exclure les points supérieurs et le second les points inférieurs à $[pq]$. Ces deux points sont calculés par deux vecteurs \vec{v} de BEZOUT vérifiant :

$$\det(\vec{u}, \vec{v}) = \pm 1$$

Dans ce cas, la généralisation qui a été présenté est immédiate.

Considérons maintenant une courbe polygonale convexe dont les sommets sont sur la grille discrète. Nous définissons la propriété de quasi-circularité de la manière suivante :

Définition 3.6 *Une courbe polygonale convexe dans \mathbb{Z}^2 est quasi-circulaire si nous pouvons séparer tous les pixels à l'intérieur de l'enveloppe convexe de cette courbe de pixels extérieurs par un cercle.*

D'après le corollaire 3.1, nous avons (voir illustration figure 3.8-(b)) :

Proposition 3.3 *Une courbe polygonale convexe $\Gamma = \{v_i\}_{i=1..n}$ est quasi-circulaire si et seulement si :*

$$\text{acd}(\{v_i\}_{i=1..n}, \{\mathcal{B}(v_i, v_{i+1})\}_{i=1..n-1}) \neq \emptyset$$

où $\mathcal{B}(v_i, v_{i+1})$ représente le point de BEZOUT associé à l'arête $[v_i, v_{i+1}]$ à l'extérieur de Γ .

Les notions d'intérieur et d'extérieur de Γ sont définies localement comme étant l'intérieur et l'extérieur de l'enveloppe convexe du morceau SCoC auquel l'arête considérée appartient.

3.3.5.2 Cercle discret et quasi-circularité

De manière similaire, nous pouvons définir une propriété de *quasi-circularité* pour des courbes discrètes. Tout d'abord, la proposition suivante présente le lien entre les cercles discrets et le problème de séparation d'ensembles de points par un cercle :

Proposition 3.4 *Une courbe discrète convexe est la discrétisation d'un arc de cercle obtenu par le processus OBQ si et seulement si nous pouvons séparer les pixels de cette courbe des pixels extérieurs par un arc de cercle.*

Dans ce qui suit, nous disons qu'un objet discret S est convexe s'il existe une région convexe R de \mathbb{R}^2 telle que (KIM, 1981; KIM et ROSENFELD, 1982) :

$$S = \mathbb{Z}^2 \cap R$$

Nous définissons une courbe discrète convexe comme étant un morceau du bord d'un objet discret convexe.

D'après la proposition 3.3, nous avons :

Théorème 3.2 *Les propriétés suivantes sont équivalentes :*

- *une courbe discrète, morceau de contour d'un objet discret convexe, est la discrétisation d'un arc de cercle euclidien*
- *la ligne polygonale $\Gamma = \{v_i\}_{i=1..n}$ définie par l'enveloppe convexe de la courbe discrète est quasi-circulaire*
- *$acd(\{v_i\}_{i=1..n}, \{\mathcal{B}(v_i, v_{i+1})\}_{i=1..n-1}) \neq \emptyset$*

La preuve de ce théorème est directement induite des propositions précédentes. Notons que l'enveloppe convexe utilisée correspond à la partie évidente de l'enveloppe convexe classique de la courbe discrète pour notre problème. Cette partie évidente correspond à la suppression de l'arête joignant les pixels extrémité dans le cas d'un arc non-fermé (voir figure 3.10).

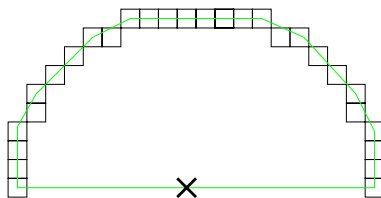


FIG. 3.10 – *L'enveloppe convexe considérée dans le théorème 3.2 ne contient pas l'arête marquée d'une croix.*

3.3.5.3 Algorithmes de reconnaissance et segmentation

Dans ce paragraphe, nous présentons différents algorithmes de reconnaissance ou de segmentation d'une courbe discrète en morceaux de cercles discrets.

Étant donnée une courbe discrète qui correspond au contour d'un objet discret, la première étape consiste en un calcul de la courbe polygonale Γ .

Si la courbe est convexe, il nous suffit de construire l'enveloppe convexe des pixels et ensuite de supprimer l'arête indésirable dans le cas où la courbe n'est pas fermée (figure 3.10). Si cette courbe n'est pas convexe, nous effectuons un pré-traitement qui consiste en une segmentation de la courbe en morceaux strictement convexes ou concaves (SCoC dans la suite). Cette pré-segmentation s'effectue grâce aux algorithmes classiques de segmentation en droites discrètes (voir paragraphe 2.2.2.5 et DEBLED-RENNESON ET AL. 2000). En effet, en analysant la segmentation en droites discrètes de la courbe, on peut trivialement extraire ces morceaux SCoC. Remarquons qu'étant donné que la segmentation en droites utilisée n'est pas optimale en nombre de morceaux, la segmentation en SCoC n'est qu'une segmentation *gloutonne* et n'offre aucune garantie d'optimalité. La figure 3.11-(a) illustre ce processus.

Ensuite, une enveloppe convexe est calculée sur chaque morceau SCoC nous permettant de construire la courbe polygonale Γ par concaténation des enveloppes convexes de chaque morceau (voir figure 3.11).

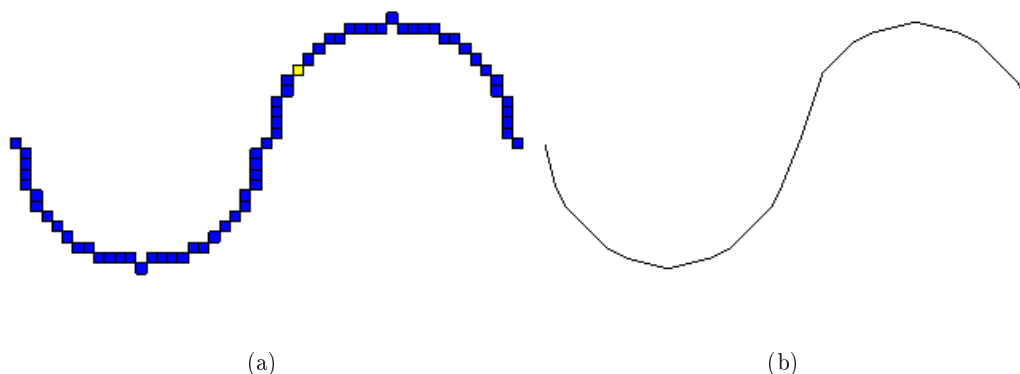


FIG. 3.11 –: Exemple de segmentation en morceaux SCoC d'une courbe discrète : (a) courbe discrète après segmentation et (b) construction de la courbe polygonale Γ .

Revenons sur le calcul de l'enveloppe convexe.

Enveloppe convexe arithmétique

En géométrie algorithmique classique, le calcul de l'enveloppe convexe de n points dans le plan se fait en $O(n \log(n))$ (voir par exemple PREPARATA et SHAMOS 1985; DE BERG ET AL. 2000). Dans le cas où les points sont issus d'une courbe discrète, des algorithmes existent pour construire une telle enveloppe en $O(n)$ (VOSS, 1993). Le coût de la segmentation en droites discrètes étant linéaire en le nombre de points, la construction de la courbe polygonale Γ à partir d'une courbe discrète quelconque s'effectue en $O(n)$ où n est le nombre de pixels de la courbe.

Cependant, pour notre calcul de l'*acd*, nous avons besoin d'avoir une écriture de chaque pente d'arête sous forme de fraction irréductible. Nous appelons un tel objet une *enveloppe convexe arithmétique*.

Une approche naïve pour le calcul d'une telle enveloppe consiste en une réduction des fractions par l'algorithme d'EUCLIDE après un algorithme classique d'enveloppe convexe. Cette solution n'est pas très intéressante sur le plan algorithmique puisque

le coût de celle-ci est en $O(n \log(t))$ où t est la taille arithmétique des fractions à résoudre (HARDY et WRIGHT, 1975). Cependant, si nous utilisons l'algorithme d'EUCLIDE étendu, nous calculons dans le même temps le vecteur de BEZOUT associé à chaque arête.

Nous pouvons cependant calculer une telle enveloppe arithmétique en $O(n)$. Cet algorithme se base sur la construction préalable d'une segmentation en droites discrètes de la courbe et sur les liens entre cette segmentation et l'enveloppe convexe. Considérons dans un premier temps une droite discrète de pente $(a, b) \in \mathbb{Z}^2$ telle que $\text{pgcd}(a, b) = 1$. L'algorithme 13 retourne les sommets inférieurs de l'enveloppe convexe au segment $[(0, 0), (b, a)]$ avec $b > a > 0$. Cet algorithme est une variante de la construction du diagramme de KLEIN (HARDY et WRIGHT, 1975) qui est elle-même basée sur l'algorithme classique de division d'EUCLIDE. En effet, les sommets de l'enveloppe convexe inférieure de ce segment se calculent à partir de la décomposition en fraction continue de $\frac{b}{a}$. Le processus utilisé pour construire ces sommets à partir de la division entière de $\frac{b}{a}$ se base sur des propriétés de symétrie associées aux droites discrètes et au diagramme de KLEIN classique (voir figure 3.12¹).

Cet algorithme nous permet donc de construire l'ensemble des sommets de l'enveloppe convexe associés à un segment de droite (les sommets supérieurs sont construits trivialement par symétrie) en $O(\log(\min(a, b)))$. À partir de ce processus, nous pouvons facilement étendre ce calcul aux segments de droites de plusieurs périodes. Il suffit dans ce cas de répéter les arêtes correspondant aux pointillés d'appui. L'utilisation de l'algorithme de VOSS (1993) aurait introduit une complexité en $O(\max(a, b))$ pour le segment considéré.

Bien évidemment, tous les vecteurs définis par deux points consécutifs de l'enveloppe convexe ont une pente donnée sous forme irréductible. De plus, avec une modification de l'algorithme 13 similaire aux modifications entre la division d'EUCLIDE et sa version étendue, nous pouvons extraire le vecteur de BEZOUT associé à chaque arête.

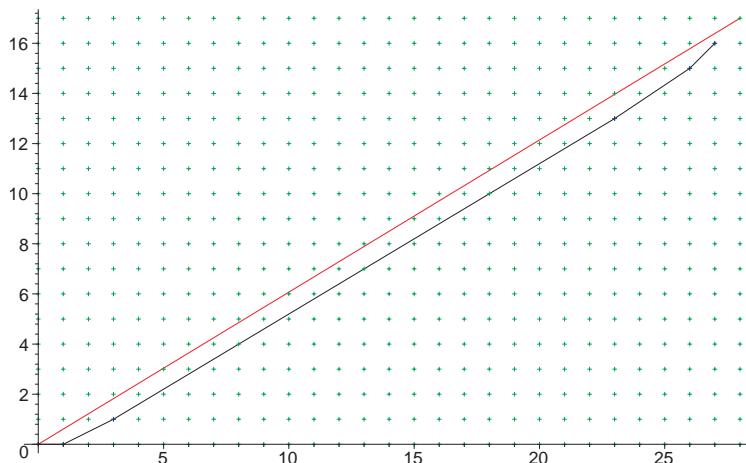


FIG. 3.12 —: Exemple de calcul d'enveloppe convexe arithmétique inférieure d'un segment de droite discrète de paramètres (17, 28).

Enfin, pour passer de ce calcul d'enveloppe convexe arithmétique des segments discrets au calcul plus général sur des bords d'objets discrets, nous utilisons une approche similaire à DEBLED-RENNESON ET AL. (2000) qui consiste à calculer une segmentation en droites discrètes, puis nous calculons les sommets et vecteurs de BEZOUT sur chaque segment avec l'algorithme présenté ci-dessous.

1. merci à J.P Reveillès pour son implémentation Maple.

Algorithme 13 Enveloppe convexe arithmétique d'un segment de droite discrète $[0, 0], [b, a]$

```

ENVELOPPE_CONVEXE_ARITHMETIQUE( $a, b$ )
1:  $a' = a, b' = b$ 
2:  $V_0 = [0, 1], V_1 = [1, 0]$ 
3:  $L_0 = V_0, L_1 = [b, a] - V_1, Liste = Vrai$ 
4: tant que  $a' > 1$  faire
5:    $q = b'/a'$  {Division entière}
6:    $r = b' - qa'$ 
7:    $b' = a', a' = r$ 
8:    $V = V_0 + qV_1, V_0 = V_1, V_1 = V$ 
9:   si  $Liste = Vrai$  alors
10:     $L_0 = L_0 \oplus \{V_1\}$  {l'opérateur  $\oplus$  est une concaténation de listes}
11:   sinon
12:     $L_1 = \{[b, a] - V_1\} \oplus L_1$ 
13:   fin du si
14:    $Liste = non(Liste)$ 
15: fin du tant que
    {Translation de  $(0, -1)$  des sommets}
16: pour point dans  $[L_0, L_1]$  faire
17:    $Liste\_Sommets = \{point - [0, 1]\} \oplus Liste\_Sommets$ 
18: fin du pour
19: retourner  $Liste\_Sommets$ 

```

Pour conclure sur ce calcul, nous avons une construction de la courbe polygonale Γ avec les points de BEZOUT associés à chaque arête en un temps borné par $O(n)$.

Reconnaissance globale de cercle

Dans un premier temps, nous nous intéressons au problème suivant : étant donnée une courbe discrète, est-ce que celle-ci est un arc de cercle discret ? Bien évidemment, si la réponse est positive, nous souhaitons avoir le domaine des centres possibles ou *acd*.

Nous supposons connue la fonction *acd*(S, T) qui permet de construire l'*acd* avec l'algorithme de PREPARATA et SHAMOS (1985) (voir annexe C). L'algorithme 14 présente le processus de reconnaissance qui consiste à construire simplement les ensembles S et T à partir d'une courbe discrète et qui retourne le domaine des solutions correspondant.

Si N représente le nombre d'arêtes de la courbe Γ , nous obtenons N points de BEZOUT et $N + 1$ sommets. Lors de l'appel *acd*(S, T) nous devons donc calculer l'intersection de $O(N^2)$ contraintes. Nous reviendrons plus tard sur l'analyse de complexité.

Segmentation en arcs de cercle

Nous cherchons maintenant à segmenter notre courbe en arcs de cercle. De plus, nous souhaitons une description complète de l'ensemble des solutions pour chaque arc (voir algorithme 15).

Nous transformons pour cela l'algorithme précédent en une version incrémentale. Nous supposons aussi que l'implémentation de la fonction *acd*(S, T) est aussi incrémentale, ce qui est le cas dans l'algorithme basé sur la dualité point-droite de PRE-

Algorithme 14 Reconnaissance globale de cercle discret (COEURJOLLY ET AL., 2001c)

RECONNAISSANCE_CERCLE_GLOBAL(C) $\{C$ est une courbe discrète}

- 1: $S = \emptyset$
- 2: $T = \emptyset$
- 3: Construction de Γ
- 4: **si** Γ est composée de plus d'un morceau SCoC **alors**
- 5: **retourner** \emptyset $\{\text{si la courbe n'est pas SCoC, ce n'est pas un morceau de cercle}\}$
- 6: **sinon**
- 7: **pour** toutes les arêtes $[v_i v_{i+1}]$ de Γ **faire**
- 8: Soit \mathcal{B} le point de BEZOUT de $[v_i v_{i+1}]$
- 9: $S = S + \{v_i, v_{i+1}\}$
- 10: $T = T + \{\mathcal{B}\}$
- 11: **fin du pour**
- 12: **retourner** $acd(S, T)$
- 13: **fin du si**

PARATA et SHAMOS (1985).

Algorithme 15 Segmentation en arcs de cercle (COEURJOLLY ET AL., 2001c)

SEGMENTATION_ARC(C) $\{C$ est une courbe discrète}

- 1: $S = \emptyset, T = \emptyset$
- 2: $Resultat = \emptyset$
- 3: Construction de Γ
- 4: **pour** chaque morceau SCoC notée C_i de Γ **faire**
- 5: **pour** toutes les arêtes $[v_j v_{j+1}]$ de C_i **faire**
- 6: Soit \mathcal{B} le point de BEZOUT de $[v_j v_{j+1}]$
- 7: $S = S + \{v_j, v_{j+1}\}$
- 8: $T = T + \{\mathcal{B}\}$
- 9: $temp_{acd} = acd(S, T)$
- 10: **si** $temp_{acd} = \emptyset$ **alors**
- 11: $S = \emptyset, T = \emptyset$ $\{\text{nouvel arc de cercle}\}$
- 12: $Resultat = \{prec_{acd}\} \oplus Resultat$ $\{\text{l'opérateur } \oplus \text{ permet de concaténer les différents } acd\}$
- 13: **sinon**
- 14: $prec_{acd} = temp_{acd}$
- 15: **fin du si**
- 16: **fin du pour**
- 17: $Resultat = \{prec_{acd}\} \oplus Resultat$
- 18: **fin du pour**
- 19: **retourner** $Resultat$

3.3.5.4 Analyse de complexité

Nous évaluons maintenant la complexité des différents algorithmes. Comme nous l'avons vu précédemment, l'élément qu'il nous faut estimer est le nombre d'arêtes N de la courbe Γ en fonction du nombre de pixels de la courbe.

Pour cela, nous reprenons le théorème de ACKETA et ŽUNIĆ (1995) :

Théorème 3.3 (ACKETA et ŽUNIĆ) *Soit une courbe convexe discrète dans une*

grille $m \times m$, le nombre maximal d'arêtes $e(m)$ de cette courbe est donné par :

$$e(m) = \frac{12}{(4\pi^2)^{1/3}} m^{2/3} + O(m^{1/3} \log(m))$$

Grâce à cette borne, nous pouvons calculer la complexité des algorithmes présentés. Pour cela, nous adaptons le théorème précédent à notre étude :

Corollaire 3.2 *Soit un cercle discret de n pixels, le nombre d'arêtes de son enveloppe convexe est en $O(n^{2/3})$*

Il nous suffit de considérer le cas des cercles discrets puisqu'ils représentent les *pires cas* pour la reconnaissance.

Reconnaissance globale

Pour le processus global, nous avons montré qu'un calcul d'intersection de N^2 contraintes est nécessaire où $N = |\Gamma|$. Or, avec le corollaire précédent et étant donné que l'algorithme de calcul de l'*acd* est en $O(|S||T| \log(|S||T|))$, le coût de l'algorithme 14 est de $O(n^{4/3} \log(n))$.

Segmentation en arcs de cercles

Étant donné que les outils utilisés pour calculer l'*acd* sont des outils classiques en géométrie algorithmique, nous pouvons utiliser toutes les optimisations qui ont été proposées pour ces derniers.

Ainsi, l'algorithme de PREPARATA et SHAMOS (1985) s'implémente sans difficulté sous forme incrémentale avec un coût logarithmique à chaque insertion de contrainte. De plus étant donné que, d'un morceau SCoC à l'autre, l'ensemble des contraintes est remis à zéro, le coût de cet algorithme est exactement le même que le précédent, à savoir $O(n^{4/3} \log(n))$ pour une courbe de n pixels.

3.3.6 Quelques résultats

Dans un premier temps, nous considérons une courbe discrète qui correspond à un cercle discret. Nous calculons la courbe polygonale Γ de cette courbe et nous souhaitons analyser le comportement de l'*acd* lors d'une reconnaissance incrémentale.

Ainsi, en fonction du nombre d'arêtes considéré de Γ , nous analysons la forme de l'*acd* avec les mesures suivantes : centre de gravité du polygone et écart type de la position des sommets. Ces mesures sont données une fois que le domaine *acd* est borné.

Prenons comme exemple un cercle discret de rayon 100. Le premier résultat obtenu est que dans tous les cas, le centre $(0, 0)$ du cercle est toujours dans le domaine des solutions. Nous remarquons aussi que la forme du polygone des solutions converge très vite vers une bonne approximation du centre du cercle euclidien sous-jacent. Le graphe de la figure 3.13 trace la distance euclidienne entre le centre de gravité de l'*acd* et le centre du cercle, en fonction du nombre d'arêtes considérées de Γ .

Le tableau 3.2 présente quelques résultats numériques sur le cercle discret de rayon 100. Nous voyons ainsi une très bonne localisation du centre exact avec peu d'arêtes de Γ . La figure 3.14 illustre la forme du domaine en fonction du nombre d'arêtes.

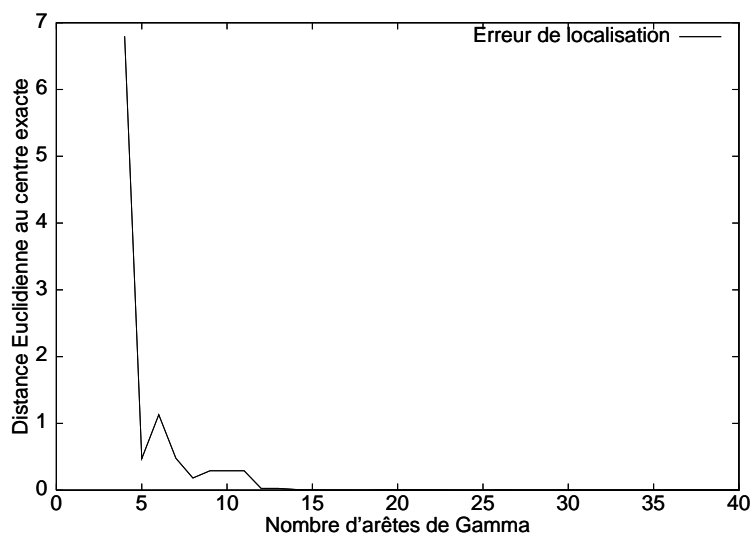


FIG. 3.13 – : Distance euclidienne entre le centre du cercle et le centre de gravité de l'acd en fonction du nombre d'arêtes de Γ considérées pour un cercle de rayon 100.

Nombre d'arêtes	\bar{x}	\bar{y}	σ_x	σ_y
5	0.338444	.328996	2.772722	17.091599
20	0.005424	0.001096	0.000383	6.653609e-05
30	0.004237	0.000741	1.511038e-05	7.513850e-06
40	0.005606	-0.000490	1.710540e-05	1.949764e-06

TAB. 3.2 – : Résultats numériques sur la reconnaissance de cercle sur un cercle discret de rayon 100.

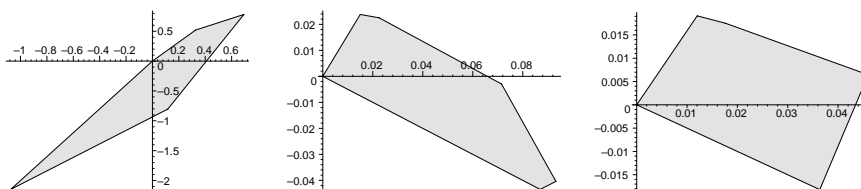


FIG. 3.14 – : Exemples d'évolution de l'acd sur un cercle de rayon 100 quand on considère un nombre croissant d'arêtes de Γ .

Les figures 3.15 et 3.16 présentent quelques exemples de segmentation de courbes discrètes. Notons que la courbe en S est construite à partir de deux arcs de cercle. Étant donné que le processus de segmentation est *glouton*, en aucun cas nous ne garantissons un nombre minimal d'arcs. C'est pour cela que la segmentation du S génère un morceau non souhaitable et que la segmentation de la parabole n'est pas symétrique.

3.3.7 Comparaison avec KOVALEVSKY (1990)

Nous terminons ce chapitre par une brève comparaison avec l'approche de KOVALEVSKY (1990). Cette approche se base aussi sur un calcul d'acd mais tous les pixels extérieurs doivent être considérés (voir figure 3.17). KOVALEVSKY présente

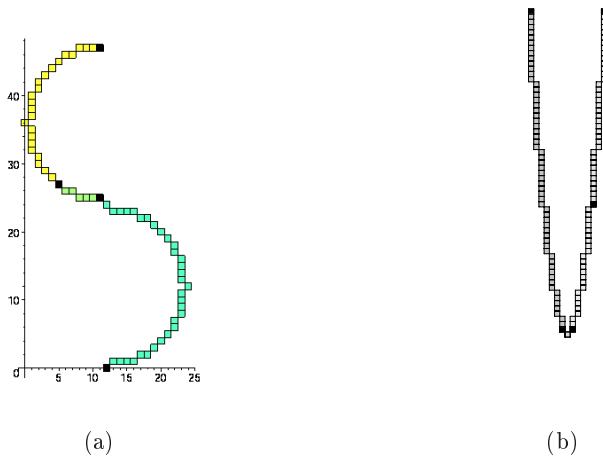


FIG. 3.15 —: Exemples de segmentation en arcs de cercles : (a) segmentation d'une courbe en S et (b) segmentation d'une parabole.

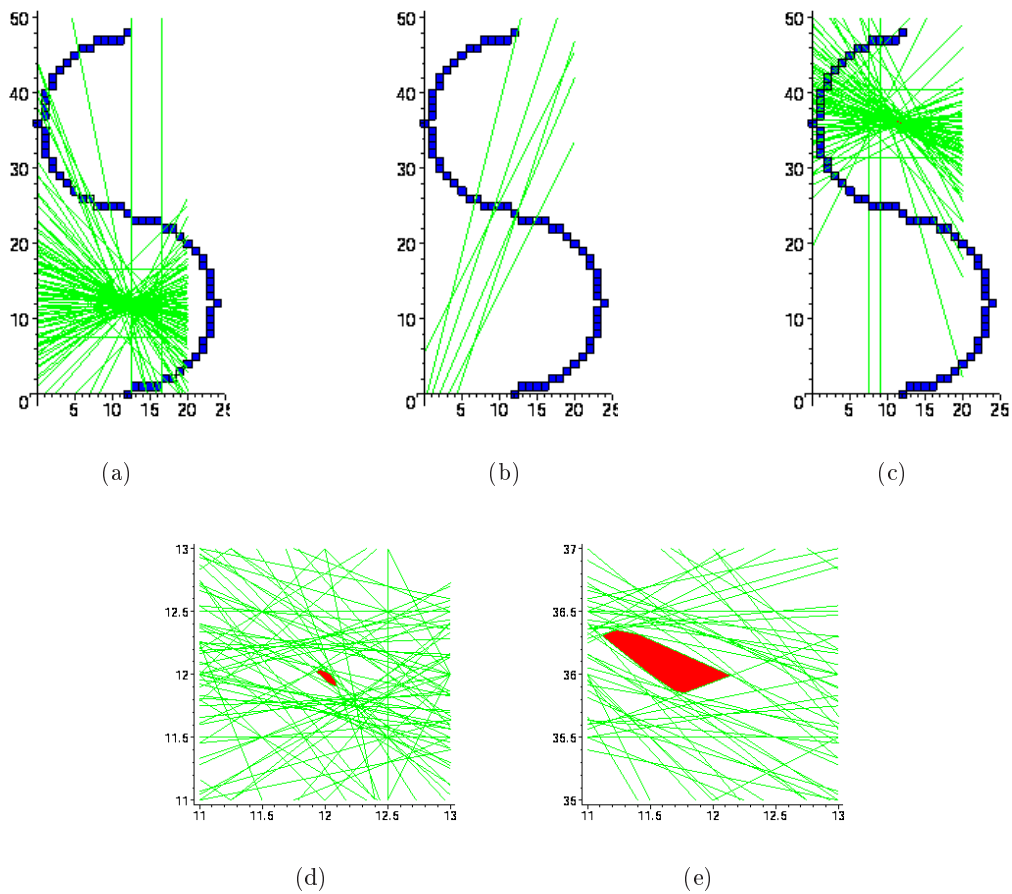


FIG. 3.16 —: Détails de la segmentation de la courbe en S : (a) – (c) illustration des contraintes utilisées pour chaque morceau et (d) – (e) zoom sur l'acd du premier et du dernier morceau. Notons que les centres exacts sont respectivement $(12, 12)$ et $(12, 36)$.

cependant une optimisation à son calcul de contraintes mais aucune borne ne peut être donnée : dans le pire cas, la complexité reste la même, c'est-à-dire $O(n^2 \log(n))$ (au mieux en utilisant PREPARATA et SHAMOS 1985). Rappelons la borne de notre algorithme qui permet en plus une segmentation en arc : $O(n^{4/3} \log(n))$.

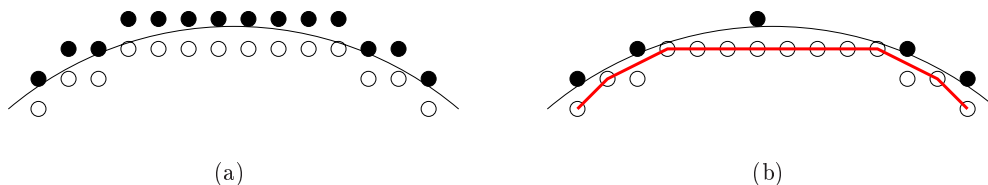


FIG. 3.17 —: *Comparaison entre notre approche et celle de KOVALEVSKY (1990) : (a) points à considérer pour l'approche de KOVALEVSKY et (b) dans notre approche, seuls les sommets de la courbe polygonale et les pixels noirs sont à considérer.*

3.4 Conclusion

Dans ce chapitre, nous avons présenté un algorithme de reconnaissance de cercles permettant la description complète du domaine des centres des cercles solutions, représenté par le polygone acd .

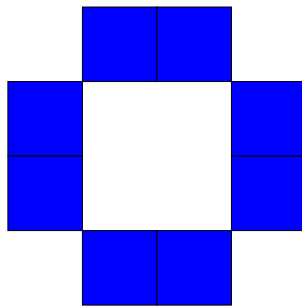
Cet algorithme se base sur des propriétés arithmétiques de la grille discrète et sur l'interprétation géométrique de théorèmes classiques en théorie des nombres (théorème de BEZOUT).

De plus, l'utilisation d'outils de géométrie algorithmique très simple (enveloppe convexe et dualité point-droite) nous a permis de proposer un algorithme de segmentation d'une courbe discrète en arcs de cercle, avec pour chaque morceau une description complète des solutions.

De nombreux problèmes connexes à ce travail apparaissent : comment segmenter une courbe discrète en un nombre minimal de sommets ? Peut-on généraliser ce travail à d'autres formes géométriques comme des ellipses ou encore des sphères en dimension 3 ?

Deuxième partie

Analyse d'objets et mesures



longueur : 9.658543
courbure moyenne : 0.02324
variance de la courbure : 0.0003422
mmhhh... ça doit être un cercle...

à moins que ce ne soit un carré ?

Chapitre 4

Métriques discrètes

Sommaire

4.1	Introduction	107
4.2	Transformée en distance	107
4.2.1	Métriques de chanfrein	107
4.2.2	Transformée en distance euclidienne	108
4.2.2.1	Approche par déplacements signés	108
4.2.2.2	Approche multi-dimensionnelle de SAITO et TORIWAKI	109
4.2.3	Diagramme de Voronoï et transformée en distance euclidienne	111
4.2.3.1	Diagramme de Voronoï et triangulation de Delaunay en géométrie algorithmique	111
4.2.3.2	Diagramme de Voronoï, triangulation de Delaunay en géométrie discrète et EDT	112
4.2.4	EDT en dimension 2 basée sur le diagramme de Voronoï discret	113
4.2.5	EDT en dimension 3 basée sur le diagramme de Voronoï discret	117
4.2.6	EDT en dimension n basée sur le diagramme de Voronoï discret	120
4.2.6.1	Analyse théorique	120
4.2.6.2	Description de l'algorithme	124
4.2.7	Analyse expérimentale	124
4.2.8	Diagramme de Voronoï discret et transformée en distance sur des grilles non-cubiques	127
4.2.9	Conclusion et perspectives	132
4.3	Squelette de forme et axe médian	134
4.3.1	Axe médian et squelette en géométrie discrète	134
4.3.2	Squelette n -dimensionnel basé sur le diagramme de Voronoï discret	136
4.3.3	Conclusion et discussion	138
4.4	Visibilité et géodésiques discrètes	143
4.4.1	Visibilité dans des domaines discrets	144
4.4.1.1	Notations et définitions	144
4.4.1.2	Domaine de visibilité	145
4.4.1.3	Algorithme de calcul de visibilité	148
4.4.2	Plus court chemin discret et métrique géodésique discrète	151

4.4.2.1	Définitions et état de l'art	151
4.4.2.2	Algorithme d'étiquetage en distance géodésique . .	152
4.4.3	Analyse expérimentale et discussion	153
4.4.4	Vers une extension en 3D	155
4.4.5	Conclusion et perspectives	158
4.5	Conclusion sur ce chapitre	158

4.1 Introduction

Dans ce chapitre, nous nous intéressons aux analyses globales des objets discrets ainsi qu'aux différentes métriques que l'on peut associer à la grille. Dans un premier temps, nous étudions le calcul de la transformée en distance euclidienne en dimension quelconque. Puis nous proposons une extraction de squelette de formes, objet géométrique très important et très utilisé pour la caractérisation d'objets. Par la suite, nous nous intéressons aux notions de visibilité et de métrique géodésique en géométrie discrète.

4.2 Transformée en distance

En analyse de formes binaires, la notion de transformée en distance est un outil classique qui est apparu avec les tous premiers algorithmes (ROSENFELD et PFALTZ, 1966, 1968). L'énoncé de cette transformation est la suivante : on étiquette chaque point de l'objet par la distance entre celui-ci et le point le plus proche du complémentaire. Une écriture plus générale de cette transformée en distance correspond à l'étiquetage de tous les points de la forme par la distance minimale à la frontière, celle-ci se trouvant entre l'objet discret et son complémentaire. En considérant la distance minimale aux points du complémentaire, nous nous plaçons dans le cadre selon lequel la frontière décrite précédemment est définie uniquement par des points du complémentaire. Dans tous les cas, cette notion est liée à la métrique utilisée. Pour des raisons algorithmiques, les premières études considéraient des métriques approchées basées sur des masques de poids appelés masques de chanfrein. Il est à noter que plus la métrique utilisée se rapproche de la métrique euclidienne, plus la description de la forme sera invariante en rotation. En d'autres termes, l'utilisation de la métrique euclidienne permet de palier l'anisotropie intrinsèque de la grille discrète.

Pour la caractérisation de forme, on extrait de cette transformée en distance un objet géométrique appelé *squelette* ou *axe médian* qui correspond au plus petit sous-ensemble de points de l'objet qui permet de reconstruire la forme en appliquant le processus de transformée en distance inverse (voir ROSENFELD et PFALTZ 1966). Ce *squelette* ou *axe médian* d'une forme correspond à la notion d'axe de symétrie généralisée (BLUM, 1967). En géométrie discrète, le terme *squelette* décrit généralement un objet topologique homotope à la forme (nous reviendrons sur ce point plus tard). Dans ce qui suit, nous construisons un objet discret que l'on appellera *squelette* mais qui ne se base que sur des propriétés géométriques (le sens de ce terme est issu de la géométrie algorithmique). Nous reviendrons sur la construction de cet objet dans le paragraphe 4.3.

Nous commençons cette partie par quelques rappels sur la transformée en distance basée sur des métriques de Chanfrein et sur la métrique euclidienne. Puis nous présentons un algorithme de calcul de transformée en distance euclidienne exacte optimal en temps pour des images n -dimensionnelles basé sur le calcul du diagramme de Voronoï discret.

4.2.1 Métriques de chanfrein

Dans un premier temps, l'idée est d'approximer la métrique euclidienne par des estimations locales de manière à obtenir des algorithmes de transformée en distance très rapides. Par exemple, ROSENFELD et PFALTZ (1966) considéraient des distances d_4

et d_8 sur la grille, plus tard ROSENFELD et PFALTZ (1968) proposaient de combiner ces distances pour se rapprocher de la distance euclidienne.

La distance d_4 (resp. d_8) est caractérisée par des poids affectés à 1 pour chaque 4-voisin (resp. pour chaque 8-voisin). Ainsi, la *boule* (ensemble de points à une distance inférieure à un certain seuil) construite sur cette métrique est un carré tourné (respectivement un carré). La figure 4.1 représente les boules de rayon 100 associées à chaque métrique avec une palette de niveaux de gris cyclique.

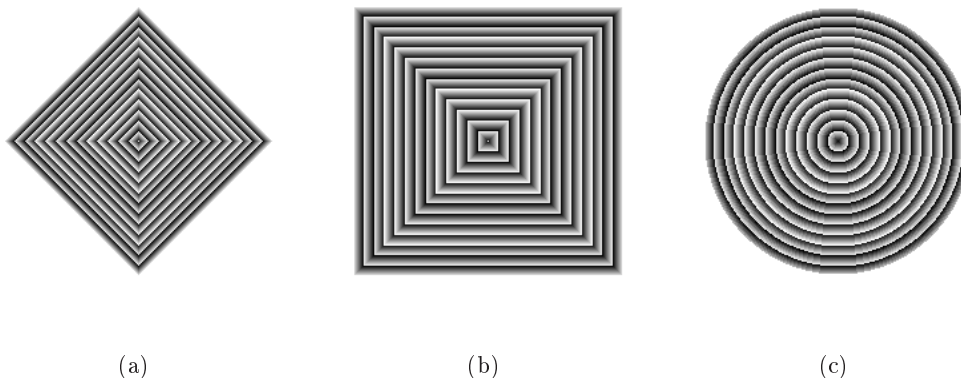


FIG. 4.1 –: Boules de rayon 100 pour les métriques : (a) d_4 (ou distance de Manhattan), (b) d_8 (ou distance de l'échiquier) et (c) distance euclidienne.

Le cadre théorique et pratique pour le calcul des poids d'un masque de chanfrein est le suivant : on commence par identifier un ensemble de déplacements élémentaires et on essaye d'optimiser les poids pour minimiser l'erreur par rapport à la distance euclidienne et pour que ceux-ci forment une distance (inégalité triangulaire ...). La description de ce processus général a été donnée par BORGEFORS (1986); VERWER (1991); THIEL (1994, 2001); REMY (2001). Ce processus général est ensuite appliqué sur les grilles par exemple tridimensionnelle (KIRYATI et KUBLER, 1995; REMY et THIEL, 2000; BORGEFORS et SVENSSON, 2001) ou pour des grilles anisotropiques (SINTORN et BORGEFORS, 2002).

L'intérêt majeur de ces approches est la simplicité de l'algorithme pour le calcul de la transformée en distance (notée WDT pour *Weighted Distance Transform*).

4.2.2 Transformée en distance euclidienne

Nous nous intéressons maintenant à la transformée en distance avec une métrique euclidienne (notée EDT pour *Euclidean Distance Transform*). Cette partie s'inscrivant dans le cadre d'une approche discrète, nous souhaitons ne manipuler que des nombres entiers et donc soit uniquement le vecteur à coordonnées entières entre le point de l'objet considéré et son plus proche point dans le complémentaire, soit la distance euclidienne au carré.

4.2.2.1 Approche par déplacements signés

Dans cette approche, DANIELSSON (1980) propose un balayage de l'image effectué en deux passes avec des opérations locales pour mettre à jour le vecteur entre le point considéré et le point du complémentaire le plus proche (on parle alors de VDT pour

Vector Distance Transform). Cet algorithme considère donc un certain voisinage (4– ou 8–voisins) mais ce choix introduit des erreurs dans l’étiquetage (voir figure 4.2 pour le cas 4–connexe, un schéma équivalent existe pour le cas 8–connexe). Ainsi, cet algorithme est linéaire en le nombre de pixels de la grille ($O(n^2)$ pour une image $n \times n$) mais l’étiquetage n’est pas sans erreur.

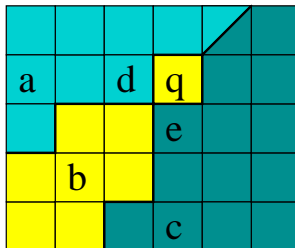


FIG. 4.2 –: *Erreur de la transformée de DANIELSSON sur un voisinage en 4–connexité (algorithme 4SED de DANIELSSON 1980.) : les points a, b et c sont des points du complémentaire et q, d et e des points de l’objet. Lors de la propagation en 4–connexité, d est plus proche de a que de b, c et e est plus proche de c, q sera noté “plus proche de a ou de c” alors que le résultat exact serait b.*

YAMADA (1984) a proposé une version parallèle de cet algorithme qui permet d’éviter ces erreurs. Par la suite, RAGNEMALM (1990) a proposé un algorithme séquentiel pour la transformée en distance euclidienne exacte mais avec une complexité qui, dans le pire cas, est en $O(n^3)$ pour une image $n \times n$. Dans cet algorithme, RAGNEMALM propose un parcours de pixels, non plus par balayage de l’image mais par propagation de fronts d’ondes. Dans le même cadre, CUISENAIRE (1999) propose un algorithme rapide basé sur une propagation en distance euclidienne. Cette méthode n’est pas sans erreur mais l’utilisation de plusieurs voisinages dans la résolution permet de réduire l’erreur d’approximation.

Une version corrigée de l’algorithme de DANIELSSON, optimale en temps a été proposé par CUISENAIRE et MACQ (1999). Dans cette approche, les auteurs effectuent un post-traitement sur la transformée en distance de DANIELSSON pour corriger les erreurs de la figure 4.2 qui sont facilement détectables dans la transformée (de manière intuitive, ce sont des *coins* dans le diagramme de Voronoï).

Des extensions pour des images 3D existent, par exemple MULLIKIN (1992) pour la version 3D de l’approche de DANIELSSON et CUISENAIRE a proposé une approche par propagation de l’algorithme précédent.

4.2.2.2 Approche multi-dimensionnelle de SAITO et TORIWAKI

SAITO et TORIWAKI (1994) ont proposé une approche basée sur la manipulation de distance au carré qui offre l’avantage d’être simple et valide quelle que soit la dimension de l’image.

Nous illustrons cet algorithme dans le cas 3D. Pour cela, nous notons \mathcal{P} l’ensemble des points composant le complémentaire de l’objet dans une image $[1 \dots m_1] \times [1 \dots m_2] \times [1 \dots m_3]$. L’objectif est de calculer la transformée en distance $S = \{s_{ijk}\}$ telle que s_{ijk} est la distance au carré entre le point (i, j, k) et le point de \mathcal{P} le plus proche. L’écriture de cette transformée est la suivante, pour tout

point q de coordonnées (i, j, k) :

$$s_q = \min_{p \in \mathcal{P}} \{dist^2(p, q)\}$$

$$= \min_{p(x,y,z) \in \mathcal{P}} \{(i-x)^2 + (j-y)^2 + (k-z)^2\}$$

Cette simple écriture ne nous permet pas de résoudre le problème mais nous laisse envisager une résolution dimension par dimension, à savoir une minimisation sur x , puis y et enfin z . Pour le cas 3D, nous obtenons l'algorithme suivant :

1. Construction, à partir de l'image source et de \mathcal{P} , d'une transformée en distance au carré selon l'axe des x . Nous notons cette transformée $G = \{g_{ijk}\}$, donnée par :

$$g_{ijk} = \min_{p(x,y,z) \in \mathcal{P}} \{(i-x)^2\}$$

2. A partir de G , nous construisons la transformée notée $H = \{h_{ijk}\}$ en utilisant une procédure de minimisation selon l'axe des y . H est donnée par :

$$h_{ijk} = \min_y \{g_{iyk} + (j-y)^2, 1 \leq y < m_2\}$$

3. Enfin, nous construisons la transformée en distance au carré finale en appliquant une autre procédure de minimisation selon l'axe des z :

$$s_{ijk} = \min_z \{h_{ijz} + (k-z)^2, 1 \leq z < m_3\}$$

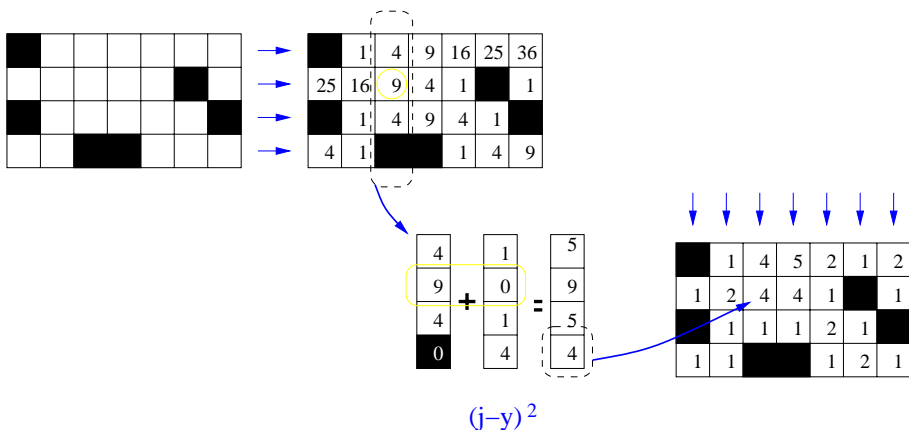


FIG. 4.3 –: Illustration de l'algorithme de SAITO et TORIWAKI en 2D : de gauche à droite, l'image source où les pixels noirs sont les pixels du complémentaire, la transformée G après la première passe, et la transformée finale après minimisation selon l'axe des y

Cet algorithme très simple est directement extensible aux images de dimension n . Dans ce cas, on applique la première phase calculant une transformée en distance au carré unidimensionnelle (étape 1 dans l'algorithme ci-dessus), puis pour chaque dimension supérieure, on minimise la transformée partielle calculée à l'étape précédente sommée avec des paraboles (fonctionnelles $(j-y)^2$ et $(k-z)^2$) dans la dimension actuelle (étape 2 et 3 de l'algorithme ci-dessus). La figure 4.3 illustre cet algorithme pour le cas 2D.

En terme de complexité, nous pouvons remarquer que pour une image 3D, la première étape s'effectue en temps linéaire en le nombre de points de l'objet. Cependant, l'opération de minimisation pour chaque dimension supérieure implique un coût supplémentaire. Ainsi, les étapes 2 et 3 ont chacune une complexité en $O(n^4)$ pour une image $n \times n \times n$. Le coût final d'une transformée en distance pour une image en dimension d est de $O(n^{d+1})$. SAITO et TORIWAKI (1994) ont cependant proposé une optimisation qui réduit la complexité en $O(Av(\sqrt{s_{ijk}})n^3)$ où $Av(\sqrt{s_{ijk}})$ représente la distance moyenne observée dans la transformée en distance de la forme.

Par la suite, HIRATA (1996) et MEIJSTER ET AL. (2000) vont indépendamment optimiser l'opération de minimisation rendant le calcul optimal, c'est-à-dire linéaire en le nombre de points de la grille et donc en $O(n^d)$ pour une image d -dimensionnelle.

4.2.3 Diagramme de Voronoï et transformée en distance euclidienne

Dans cette partie, nous proposons une autre approche pour le calcul de la transformée en distance euclidienne basée sur la construction du diagramme de Voronoï discret. Nous présentons un algorithme qui est aussi optimal en temps pour des images n -dimensionnelles. Cet algorithme construit une structure géométrique plus complexe qui permet de proposer des algorithmes de transformée en distance sur des grilles anisotropes ainsi que sur des grilles d'hexagones. De plus, nous verrons dans la partie 4.3 comment cette structure permet un calcul direct de l'axe médian de la forme.

4.2.3.1 Diagramme de Voronoï et triangulation de Delaunay en géométrie algorithmique

En géométrie algorithmique (non discrète), le diagramme de Voronoï d'un ensemble de sites $\mathcal{P} = \{p_i\} \subset \mathbb{R}^n$ est une partition de l'espace en cellules, chacune étant associée à un site avec la propriété suivante : un point q appartient à la cellule \mathcal{C}_{p_i} associée au site p_i si et seulement si $dist(q, p_i) < dist(q, p_j)$ ¹ pour tous les points $p_j \in \mathcal{P}$ et $i \neq j$ (voir figure 4.4-(a) pour un diagramme dans \mathbb{R}^2). Pour être plus précis, le diagramme de Voronoï est une partition de \mathbb{R}^n en cellules au sens des complexe simpliciaux (décomposition en points, arêtes, faces, hyper-faces, . . . , cellules de dimension n).

Ce diagramme a été très étudié et des bornes sur la complexité des algorithmes de construction existent. Par exemple en 2D, la borne inférieure est en $\Omega(|\mathcal{P}| \log |\mathcal{P}|)$ (voir PREPARATA et SHAMOS 1985).

Si l'on considère un ensemble de sites en *position générale*, c'est-à-dire tels qu'il n'existe aucun quadruplet de sites cocycliques, de nombreuses propriétés sur ce graphe peuvent être prouvées (les lecteurs pourront se référer par exemple à PREPARATA et SHAMOS 1985 , BOISSONNAT et YVINEC 1995 ou encore à DE BERG ET AL. 2000). Par exemple, chaque sommet du diagramme est incident à exactement trois cellules.

Si nous considérons le graphe dual de ce diagramme, c'est-à-dire le graphe tel que les sommets sont des sites, et les arêtes des adjacences entre cellules du diagramme, nous construisons alors une triangulation des sites qui est appelée *triangulation de Delaunay* (voir figure 4.4-(b)). Cette triangulation a aussi de nombreuses propriétés en terme *d'aspect* des triangles générés. Plus formellement, si on trie tous les angles

1. $dist(p, q)$ correspond à la distance euclidienne.

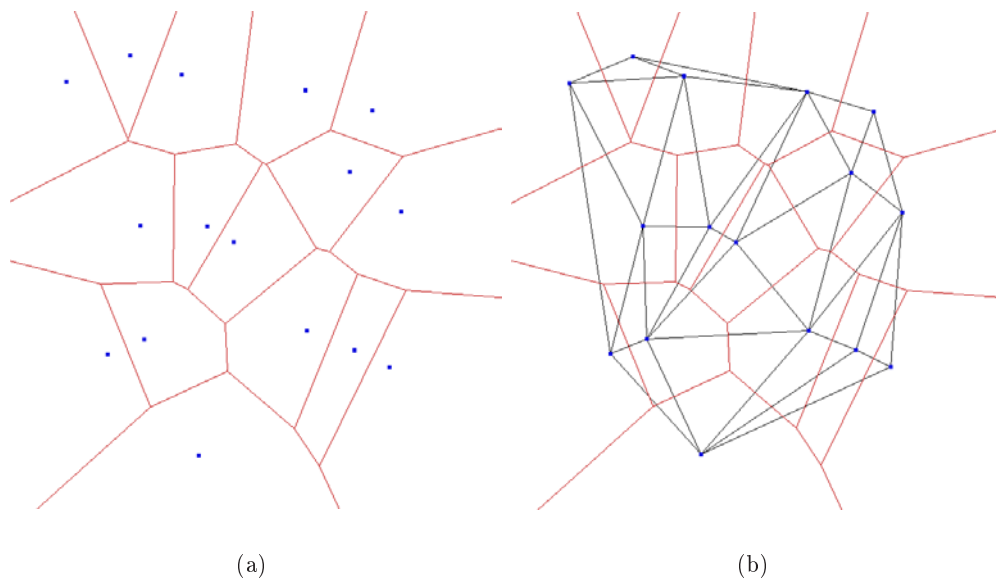


FIG. 4.4 –: (a) diagramme de Voronoï dans le plan et (b) triangulation de Delaunay (capture d'écran de VoroGlide, ICKING ET AL. 2001).

des triangles d'une triangulation \mathcal{T} des sites, dans un tableau $\mathcal{A}(\mathcal{T})$, la triangulation de Delaunay noté \mathcal{D} est minimale, c'est-à-dire $\mathcal{A}(\mathcal{D}) \preceq \mathcal{A}(\mathcal{T})$ en considérant l'ordre lexicographique pour la relation \preceq .

4.2.3.2 Diagramme de Voronoï, triangulation de Delaunay en géométrie discrète et EDT

Nous définissons une version discrète très simple du diagramme de Voronoï et de la triangulation de Delaunay qui correspond au plongement de ces structures géométriques euclidiennes dans l'espace discret.

Définition 4.1 (Diagramme de Voronoï discret) Soit $P = \{p_i\}$ un ensemble de sites de l'espace discret, le diagramme de Voronoï discret est une décomposition de l'espace discret en sous-ensembles $\{c_i\}$, tels que chaque point de c_i soit plus proche de p_i que de chacun des $\{p_j\}$ avec $j \neq i$.

Dans ce qui suit, nous ne nous intéressons pas à la topologie associée au diagramme de Voronoï mais uniquement à sa construction. Cette définition ne donne pas un diagramme unique, en effet, dans la cas où un point est équidistant de deux sites, il est arbitrairement associé à un des deux sites. Par contre, lorsqu'un point discret est associé à un site p_i , nous savons que ce point appartient à l'adhérence de la cellule c_i dans le diagramme euclidien des sites P .

A partir de la définition 4.1, nous pouvons définir une triangulation de Delaunay discrète basée sur les notions de droites discrètes.

Définition 4.2 (Triangulation de Delaunay discrète) Soit $P = \{p_i\}$ un ensemble de points discrets, la triangulation de Delaunay discrète est telle que ses sommets sont les points $\{p_i\}$ et un segment discret existe, dans l'espace considéré, entre deux sommets si ces deux sommets engendrent des cellules adjacentes dans le diagramme de Voronoï discret associé à P .

L'utilisation du diagramme de Voronoï pour le calcul de la transformée en distance en analyse d'image été montrée par BREU ET AL. (1995). Si nous considérons comme ensemble de sites P , l'ensemble des points discrets du complémentaire de l'objet discret considéré, le diagramme de Voronoï discret de P nous donne, en tout point de l'objet, le point de P qui lui est le plus proche. Ainsi, pour obtenir la transformée en distance, il suffit d'effectuer un post-traitement sur le diagramme en étiquetant tous les points discrets par la distance au point de P le plus proche donné par le diagramme. Ce post-traitement se fait en temps linéaire en le nombre de points discrets de l'objet.

Résoudre le problème du calcul du diagramme de Voronoï permet donc de résoudre le problème du calcul de la transformée en distance euclidienne. Nous présentons dans le paragraphe suivant le calcul de transformée en distance basé sur le diagramme de Voronoï proposé par BREU ET AL. pour le cas d'une grille régulière en dimension 2 pour ensuite l'étendre en nD et sur des grilles non nécessairement cubiques.

4.2.4 EDT en dimension 2 basée sur le diagramme de Voronoï discret

BREU ET AL. (1995) ont proposé un algorithme très élégant et très efficace pour le calcul du diagramme de Voronoï discret et donc pour le calcul de la transformée en distance euclidienne sans erreur dans le cas d'une image binaire 2D notée $[0 \dots m_1] \times [0 \dots m_2]$.

Dans un premier temps, cet algorithme effectue un balayage des lignes de l'image de $y = 0$ à $y = m_2$. A chaque étape $y = r$ de l'algorithme, BREU ET AL. maintiennent une structure de données qui contient le diagramme de Voronoï discret \mathcal{V}_{P_r} des sites P_r . Ces sites correspondent aux points de P dont l'ordonnée est inférieure ou égale à r (voir figure 4.5 pour les notations de BREU ET AL.).

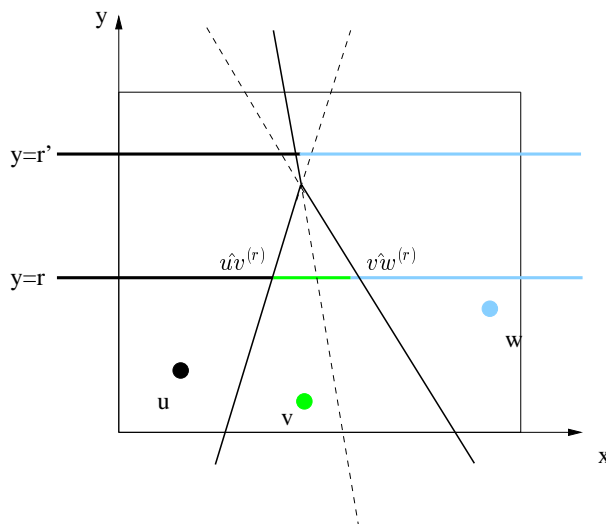


FIG. 4.5 —: Notations de BREU ET AL. : u , v et w sont des sites appartenant à P_r et $P_{r'}$, $\hat{u}v^{(r)}$ et $\hat{v}w^{(r)}$ sont les points intersections de la médiatrice de $[uv]$ et $[vw]$ respectivement, et de la droite $y = r$. Dans cet exemple, nous avons $\mathcal{L}_r = \{u, v, w\}$ et $\mathcal{L}_{r'} = \{u, w\}$ (voir le texte pour les explications).

A la fin de ce parcours, un autre balayage de $y = m_2$ à $y = 0$ est effectué pour

fusionner les résultats et ainsi construire le diagramme de Voronoï à partir de diagrammes partiels (voir figure 4.6). Cette technique correspond à une réécriture, pour le cas discret, des algorithmes de construction du diagramme de Voronoï par balayage de ligne (*plane sweep algorithm* dans la littérature), voir par exemple DE BERG ET AL. (2000).

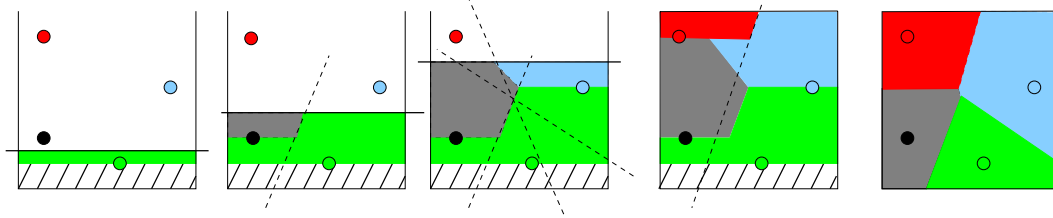


FIG. 4.6 –: Balayage de l'image en y croissant (4 premières images) avec les diagrammes de Voronoï partiels. Chaque cellule a la couleur du site auquel elle correspond, la partie hachurée correspond aux pixels non-étiquetés et les droites en pointillés sont des médiatrices entre sites.

Dans la suite, nous réutilisons les notations de BREU ET AL. : on note \mathcal{L}_r le sous-ensemble de sites P_r dont les cellules intersectent la droite $y = r$ (voir figure 4.5). Ainsi, le cœur de l'algorithme réside en un calcul efficace des sites de \mathcal{L}_r à partir des sites de \mathcal{L}_{r-1} et des nouveaux sites présents dans la ligne $y = r$ de l'image.

Cette propagation efficace est basée sur deux observations : la première est qu'il ne peut y avoir dans \mathcal{L}_r qu'un seul site par abscisse. Dans le cas contraire, la médiatrice issue des deux sites de même abscisse, soit n'intersectera pas la droite $y = r$, soit sera confondue avec celle-ci. La seconde observation est que si les sites de P_r sont triés selon leur abscisse, cet ordre est alors préservé sur la droite $y = r$ pour les cellules issues des sites de \mathcal{L}_r .

La première observation a pour conséquence que la liste \mathcal{L}_r est telle que :

$$|\mathcal{L}_r| \leq m_1$$

La seconde nous dit que les cellules de \mathcal{L}_{r-1} apparaîtront (si elles sont conservées) dans le même ordre dans \mathcal{L}_r .

Avant de détailler l'algorithme, nous présentons les conditions pour qu'un site de \mathcal{L}_{r-1} ne soit plus présent dans la liste \mathcal{L}_r . Considérons pour cela trois sites $u, v, w \in \mathbb{Z}^2$ de l'image tels que² $u_x < v_x < w_x$. On dira que le site v est *caché par u et w le long de la droite $y = r$* si la cellule associée à v n'intersecte pas la droite $y = r$ alors que c'est le cas pour les cellules associées à u et w . Cette propriété est illustrée dans la figure 4.5 pour la droite $y = r'$.

En se basant sur cette définition, nous pouvons formaliser l'approche de BREU ET AL. avec la proposition suivante :

Proposition 4.1 *Étant donné un site v d'un ensemble \mathcal{P} de sites et une droite l , v n'appartient pas à la liste \mathcal{L}_l si et seulement si il existe au moins une paire de sites u et w de $\mathcal{P} \setminus \{v\}$ tels que u et w cachent v le long de l .*

Preuve : Cette proposition est illustrée par la figure 4.7. L'idée est la suivante : si la cellule associée à v n'intersecte pas l , alors on trouve une paire de sites qui cachent v sur l en translatant l jusqu'au sommet de C_v le plus proche (droite

2. dans la suite, les coordonnées d'un point p sont notées (p_x, p_y) si $p \in \mathbb{R}^2$ et (p_x, p_y, p_z) si $p \in \mathbb{R}^3$

l' dans la figure 4.7). Ainsi, si les sites sont en position générale (il n'existe pas quatre points cocycliques), il n'existe que deux sites adjacents à ce sommet de C_v , ici u et w , et ces deux sites cachent v sur l .

Si plusieurs sites sont adjacents au sommet de C_v considéré, il suffit d'en choisir deux et ceux-ci vérifient la propriété de masquage de v .

La preuve de la proposition dans l'autre sens de l'équivalence logique est triviale : si u et w cachent v , alors $C_v \cap l = \emptyset$ par définition. \square

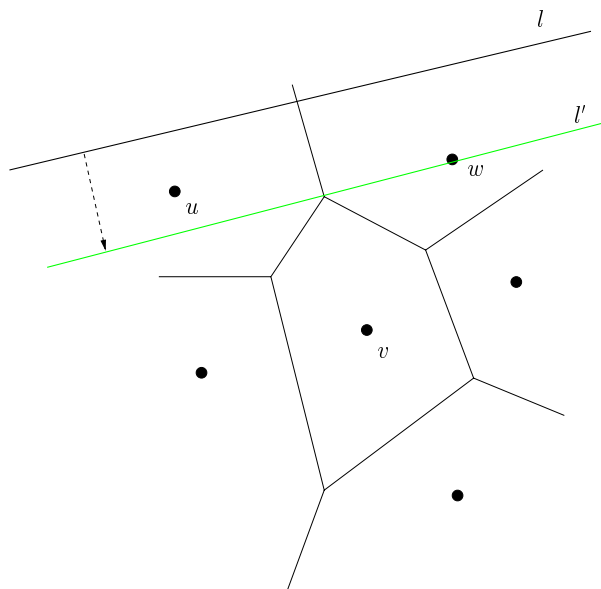


FIG. 4.7 —: Illustration de la preuve : C_v n'intersecte pas l et on calcule une paire de sites cachant v en translatant la droite l vers un sommet de C_v . Les sites adjacents à ce sommet du diagramme sont solutions du problème.

En d'autres termes, chaque *disparition* de sites dans les listes \mathcal{L}_r est dirigée par le fait qu'un site est caché par deux autres sites. Il nous faut donc un prédicat permettant de détecter ces cas.

Si on note $\hat{u}v^{(r)}$ l'intersection entre la médiatrice de $[uv]$ et la droite $y = r$, v est *caché* par u et w (avec toujours $u_x < v_x < w_x$) si et seulement si l'abscisse de $\hat{u}v^{(r)}$ est supérieure à l'abscisse de $\hat{v}w^{(r)}$. Quand il n'y a pas d'ambiguïté, et pour simplifier les notations, on écrira $\hat{u}v$ plutôt que $\hat{u}v^{(r)}$. Formellement, on a donc :

$$\hat{u}v_x \geq \hat{v}w_x \quad (4.1)$$

Si nous détaillons les coordonnées de chaque élément, $\hat{u}v$ est donné par :

$$\|u - \hat{u}v\|^2 = (u_x - \hat{u}v_x)^2 + (u_y - r)^2 \quad (4.2)$$

$$\|v - \hat{u}v\|^2 = (v_x - \hat{u}v_x)^2 + (v_y - r)^2 \quad (4.3)$$

Finalement, l'équation 4.1 est équivalente à (BREU ET AL., 1995) :

$$(w_x - v_x)(v_x^2 - 2r(v_y - u_y) + v_y^2 - u_y^2) \geq (v_x - u_x)(w_x^2 - v_x^2 - 2r(w_y - v_y) + w_y^2 - v_y^2) \quad (4.4)$$

Nous pouvons donc écrire maintenant un prédicat `CACHÉ_PAR(U,V,W,R)` qui permet de décider si u et w cachent v sur $y = r$ (voir algorithme 16).

Algorithme 16 Prédicat CACHÉ_PAR de BREU ET AL. (1995)

CACHÉ_PAR(u, v, w, r) $\{v$ est-il caché par u et w sur la droite $y = r$? $\}$
 1: **retourne** $((w_x - v_x)(v_x^2 - u_x^2 - 2r(v_y - u_y) + v_y^2 - u_y^2) \geq (v_x - u_x)(w_x^2 - v_x^2 - 2r(w_y - v_y) + w_y^2 - v_y^2))$

Avec ce prédicat et la seconde observation de BREU ET AL., nous pouvons maintenant présenter le cœur de l'algorithme qui est la procédure de suppression des sites d'une liste de sites candidats susceptibles d'appartenir à la liste \mathcal{L}_r étant donnée une droite $y = r$. Dans la suite nous notons $Candidat_r$ cette liste de sites possibles. En pratique, cette liste est composée des sites de \mathcal{L}_{r-1} et des sites présents sur la droite $y = r$. Autrement dit :

$$Candidat_r = \mathcal{L}_{r-1} \cup \{p \in P \mid p_y = r\}$$

Si les sites de $Candidat_r$ sont triés par abscisse croissante, il suffit d'appliquer le prédicat CACHÉ_PAR de manière gloutonne avec une structure de pile nous permettant de dépiler les sites qui seraient cachés par d'autres (voir algorithme 17). Nous obtenons ainsi un algorithme dont le coût est linéaire en le nombre de sites dans $Candidat_r$.

Remarquons que cet algorithme est similaire à celui utilisé par GRAHAM pour le calcul d'enveloppe convexe (voir CORMEN ET AL. 1990; DE BERG ET AL. 2000).

Algorithme 17 Procédure de suppression de sites de BREU ET AL. (1995)

SUPPRESSION_SITES($Candidat_r, r$)
 1: $\mathcal{L}_r[1] = Candidat_r[1]$
 2: $\mathcal{L}_r[2] = Candidat_r[2]$
 3: $k=2, l=3, c=|Candidat_r|$
 4: **tant que** $l \leq c$ **faire**
 5: $w = Candidat_r[l]$
 6: **tant que** $k \geq 2$ et CACHÉ_PAR($\mathcal{L}_r[k-1], \mathcal{L}_r[k], w, r$) **faire**
 7: $k = k - 1$
 8: **fin du tant que**
 9: $k = k + 1, l = l + 1$
 10: $\mathcal{L}_r[k] = w$
 11: **fin du tant que**

Or, d'après la première observation, la liste \mathcal{L}_r a une taille bornée par m_1 et par conséquent c'est aussi le cas pour la liste $Candidat_r$. De plus, la construction de cette liste à partir de \mathcal{L}_{r-1} se fait en $O(m_1)$.

Le passage de \mathcal{L}_{r-1} vers \mathcal{L}_r se fait donc en $O(m_1)$. Comme uniquement deux balayages sont nécessaires, nous obtenons une complexité finale pour la construction du diagramme de Voronoï discret et de l'EDT en $O(m_1 m_2)$, c'est-à-dire linéaire en le nombre de pixels de l'image ; ce qui est optimal pour le problème de la transformée en distance.

Récemment, GUAN et MA (1998) ont présenté une optimisation de cet algorithme en considérant une approche basée sur des listes, et ramènent la complexité en $O(\sqrt{m_1 m_2} t)$ où t représente le nombre de sites mais l'algorithme nécessite toujours une étape d'initialisation en $O(m_1 m_2)$.

Dans le paragraphe suivant, nous détaillons une version tridimensionnelle de cet algorithme avant de l'étendre finalement à des images en nD .

4.2.5 EDT en dimension 3 basée sur le diagramme de Voronoï discret

Dans la suite, une image \mathcal{I} en n dimensions sera un sous-ensemble de \mathbb{Z}^n noté $[1, m_1] \times \dots \times [1, m_n]$. Nous formalisons tout d'abord la notion de *tube* dans une grille de \mathbb{Z}^n . La notion de tube est une extension en nD des lignes, colonnes et hauteurs en 2D/3D.

Définition 4.3 (tube selon la $k^{\text{ème}}$ dimension dans une image en dimension n)

Un tube est un ensemble de m_k points du domaine $[1, m_1] \times \dots \times [1, m_n]$ tel que les coordonnées des points ne diffèrent que dans la k dimension. Ce tube est noté :

$$T_{i_1, \dots, i_{k-1}, \bullet, i_{k+1}, \dots, i_m} = \{(i_1, \dots, i_{k-1}, i_k, i_{k+1}, \dots, i_n) \in \mathcal{I} \mid 1 \leq i_k \leq m_k\}$$

Dans l'approche multidimensionnelle de SAITO et TORIWAKI, les fusions de distances sont calculées le long de *tubes* orthogonaux.

L'idée de la version 3D de l'algorithme de transformée en distance basée sur le diagramme de Voronoï discret est la suivante : utiliser la décomposition en tubes et construire les diagrammes partiels dans chaque dimension.

Dans SAITO et TORIWAKI (1994), les auteurs ont présenté un algorithme dérivé de leur calcul de transformée en distance pour construire un tel diagramme. Dans notre approche, nous proposons une construction basée sur l'approche de BREU ET AL. permettant ainsi une construction en temps optimal ainsi qu'une extensibilité pour des grilles non-cubiques.

Cette approche se décompose donc en trois étapes, une par dimension :

Étape 1 : lors de cette première étape, nous calculons un diagramme de Voronoï de dimension 1 dans les tubes parallèles à l'axe des x . En d'autres termes, nous étiquetons tous les pixels de l'objet par le plus proche pixel du complémentaire ayant les mêmes coordonnées en y et en z . Si un point de l'objet est équidistant de deux sites, nous choisissons arbitrairement un représentant, ce choix n'aura aucune importance pour la transformée en distance.

Ce calcul de diagramme de Voronoï unidimensionnel se fait en deux parcours de chaque tube et donc le coût total de cette première étape est en $O(m_1 m_2 m_3)$ (voir figure 4.8-(b)). Dans le cas particulier où il n'y a pas de site de même y et de même z , nous utilisons une étiquette spéciale notée \emptyset .

Étape 2 : l'objectif est ici de construire des diagrammes de Voronoï discrets dans chaque coupe (x, y) de l'image 3D. Pour cela, nous considérons un tube T défini par :

$$T_{r, \bullet, t} = \{(x, y, z) \in \mathcal{I} \mid x = r, z = t, y \in [1 \dots m_2]\}$$

Chaque case (r, y, t) de ce tube contient les coordonnées du site le plus proche dans chaque ligne (au sens axe des x). Nous appellerons dans la suite, les sites présents dans ce tube des *candidats*. En d'autres termes, ces sites sont utilisés dans les diagrammes 1D et sont candidats pour les diagrammes 2D définis dans les coupes.

Ainsi, nous avons une liste de candidats telle que $|Candidat| \leq m_2$ et nous pouvons faire les mêmes remarques que BREU ET AL. :

- nous n'avons qu'un seul candidat par y (par construction à l'étape 1) ;

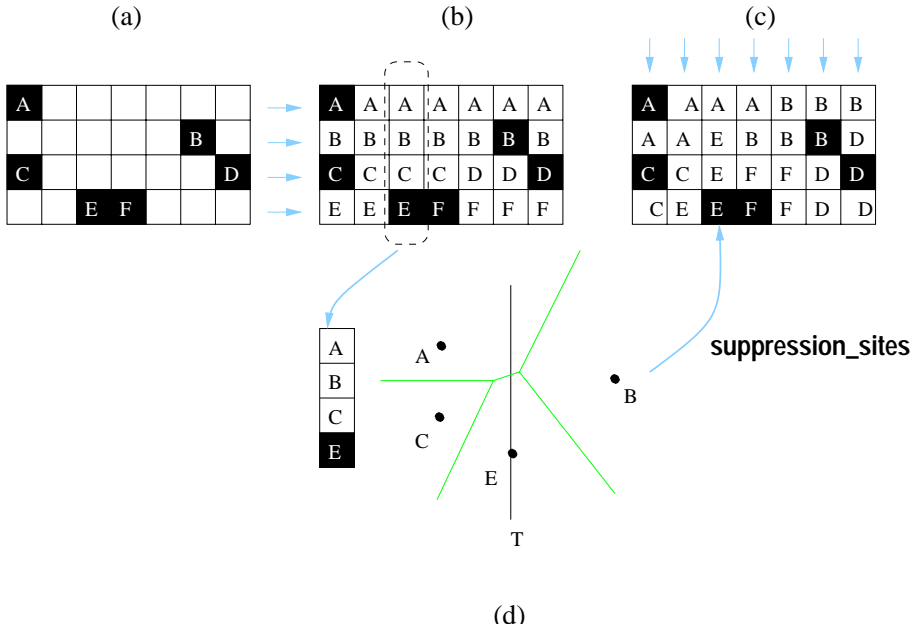


FIG. 4.8 – : Illustration du calcul 2D : les sites sont étiquetés par une lettre et on souhaite étiqueter les pixels de l'objet par le site le plus proche. (a) l'image binaire (les cases noires représentent les sites), (b) résultat après les deux premiers balayages en ligne, (c) résultat après suppression des sites dans les colonnes. (d) détail de la procédure de suppression : C est caché par A et E, B est caché par A et E le long de la colonne considérée.

- l'ordre selon les ordonnées des sites dont les cellules intersectent le tube dans le diagramme final est le même que celui des candidats du tube.

Nous pouvons donc appliquer la procédure de suppression des sites de BREU ET AL. donné par l'algorithme 17, de manière à supprimer de la liste *Candidat* les sites dont les cellules vont disparaître du tube T (voir figure 4.8). Il nous faut cependant présenter un nouveau prédicat `CACHÉ_PAR_2D` étant donné que la suppression se fait verticalement et non horizontalement comme dans l'algorithme initial (voir algorithme 18).

Algorithme 18 Nouveau prédicat `CACHÉ_PAR_2D`

`CACHÉ_PAR_2D`(u, v, w, r) $\{v$ est-il caché par u et w sur la droite $x = r$? $\}$
 1: **retourne** $((w_y - v_y)(v_y^2 - u_y^2 - 2r(v_x - u_x) + v_x^2 - u_x^2) \geq (v_y - u_y)(w_y^2 - v_y^2 - 2r(w_x - v_x) + w_x^2 - v_x^2))$

Nous appliquons donc l'algorithme 17 avec ce prédicat et nous construisons ainsi des diagrammes de Voronoï discrets dans chaque coupe de l'image. Étant donné que cette suppression sur un tube se fait en temps linéaire, c'est-à-dire en $O(m_2)$ selon nos notations, le coût global de cette étape est donc en $O(m_1 m_2 m_3)$.

Étape 3 : l'objectif de cette étape est de fusionner les différents diagrammes de Voronoï discrets 2D de chaque coupe pour construire le diagramme 3D.

Nous considérons donc le tube T' défini dans l'image par :

$$T'_{r,t,\bullet} = \{(x, y, z) \in \mathcal{I} \mid x = r, y = t, z \in [1 \dots m_3]\} \quad (4.5)$$

A chaque point (r, t, z) de ce tube, nous avons les coordonnées du site le plus proche du voxel (r, t, z) parmi les sites appartenant au plan parallèle à Oxy et de hauteur z . Les sites du tube T' définissent alors une liste de candidats pour le diagramme partiel 3D le long de T' .

Nous pouvons faire les mêmes observations que BREU ET AL. pour le cas 3D :

- nous n'avons qu'un seul candidat par z (c'est-à-dire un par coupe) ;
- l'ordre selon les coordonnées en " z " des sites dont les cellules intersectent le tube dans le diagramme final est le même que celui des candidats du tube.

L'idée est encore ici d'appliquer une suppression de sites sur ce tube (voir figure 4.9). Avant cela, il nous faut détailler le prédicat `CACHÉ_PAR_3D` qui sera utilisé. Celui-ci prend en arguments trois points $u, v, w \in \mathbb{Z}^3$ et le paramètre du tube 3D $(x = r, y = t)$. Il décide si le site v est *caché* par u et w sur le long de la droite $(x = r, y = t)$.

De la même façon que les équations (4.2), (4.3) et (4.4), nous avons :

$$\|u - \hat{u}v\|^2 = (u_x - r)^2 + (u_y - t)^2 + (u_z - \hat{u}v_z)^2 \quad (4.6)$$

$$\|v - \hat{u}v\|^2 = (v_x - r)^2 + (v_y - t)^2 + (v_z - \hat{u}v_z)^2 \quad (4.7)$$

Et donc :

$$\hat{u}v_z = \frac{v_z^2 - u_z^2 + (v_x - r)^2 + (v_y - t)^2 - (u_x - r)^2 - (u_y - t)^2}{2(v_z - u_z)} \quad (4.8)$$

Nous pouvons maintenant écrire le prédicat qui sera utilisé pour la suppression 3D (voir algorithme 19).

Algorithme 19 Prédicat `CACHÉ_PAR_3D`

`CACHÉ_PAR_3D`(u, v, w, r, t)

{ v est-il caché par u et w sur la droite $(x = r, y = t)$?}

- 1: **retourne** $((w_z - v_z)(v_z^2 - u_z^2 + (v_x - r)^2 + (v_y - t)^2 - (u_x - r)^2 - (u_y - t)^2) \geq (v_z - u_z)(w_z^2 - v_z^2 + (w_x - r)^2 + (w_y - t)^2 - (v_x - r)^2 - (v_y - t)^2))$
-

En changeant uniquement le prédicat de l'algorithme 17, nous réutilisons celui-ci et obtenons ainsi un algorithme linéaire en la taille du tube, c'est-à-dire en $O(m_3)$, pour supprimer les sites non présents dans le diagramme final.

En appliquant ce processus pour tous les tubes, nous avons une complexité en $O(m_1 m_2 m_3)$ pour cette étape.

Coût global Étant donné que chaque étape de l'algorithme se fait en $O(m_1 m_3 m_3)$, le coût global du calcul du diagramme de Voronoï discret, et donc de la transformée en distance euclidienne 3D, se fait en $O(m_1 m_3 m_3)$ ce qui est donc linéaire en le nombre de voxels et optimal pour le problème.

L'algorithme global pour le cas 3D est présenté dans l'algorithme 20.

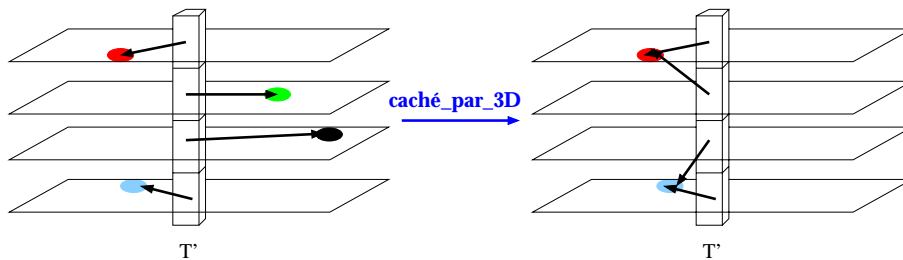


FIG. 4.9 – : Illustration de l'étape 3 : dans chaque coupe, les voxels du tube sont étiquetés par le site le plus proche dans la coupe et on utilise la suppression de sites 3D pour fusionner les diagrammes.

Algorithme 20 Diagramme de Voronoï discret en 3D

DIAGRAMME_VORONOÏ_DISCRET_3D(\mathcal{T})

{Étape 1 : selon les lignes (axe Ox)}

- 1: **pour** j de 1 à m_2 **faire**
- 2: **pour** k de 1 à m_3 **faire**
- 3: Calcul du diagramme de Voronoï unidimensionnel dans le tube $T_{\bullet,j,k}$
- 4: **fin du pour**
- 5: **fin du pour**

{Étape 2 : selon les colonnes (axe Oy)}

- 6: **pour** i de 1 à m_1 **faire**
- 7: **pour** k de 1 à m_3 **faire**
- 8: Appliquer la suppression de sites dans le tube $T_{i,\bullet,k}$
avec le prédicat CACHÉ_PAR_2D
- 9: **fin du pour**
- 10: **fin du pour**

{Étape 3 : selon les hauteurs (axe Oz)}

- 11: **pour** i de 1 à m_1 **faire**
 - 12: **pour** j de 1 à m_2 **faire**
 - 13: Appliquer la suppression de sites dans le tube $T_{i,j,\bullet}$
avec le prédicat CACHÉ_PAR_3D
 - 14: **fin du pour**
 - 15: **fin du pour**
-

4.2.6 EDT en dimension n basée sur le diagramme de Voronoï discret

Dans ce paragraphe, nous généralisons l'approche proposée en 3D pour des images n -dimensionnelles permettant le calcul du diagramme de Voronoï discret et de la transformée en distance euclidienne en temps optimal. Nous prouvons aussi les différentes observations faites précédemment dans le cas nD .

4.2.6.1 Analyse théorique

Introduisons tout d'abord quelques notations. Nous considérons par la suite un diagramme de Voronoï noté $\mathcal{V}_{\mathcal{P}}$ en dimension n , c'est-à-dire $\mathcal{P} \subset \mathbb{R}^n$. Nous considérons

aussi une droite l dans \mathbb{R}^n et on note Π_l la projection orthogonale de points de \mathbb{R}^n sur l . Sur cette droite, nous allons considérer une origine, notée O , et un vecteur unitaire de manière à introduire un ordre des points de l . Nous noterons aussi $\{p_i\}_{1..n}$ les sites de \mathcal{P} et $\mathcal{Q} = \{q_i\} \subseteq \mathcal{P}$ l'ensemble des sites de \mathcal{P} dont la cellule associée intersecte la droite l . Étant données les propriétés de convexité des cellules de Voronoï, on dispose également d'un ordre des cellules de \mathcal{Q} le long de la droite l : l'intersection d'un convexe avec une droite est un segment. A part dans les cas *dégénérés*, ces segments sont totalement ordonnés.

Pour prouver les observations faites en 2D et en 3D (dans les étapes 2 et 3), il nous faut montrer les deux points suivants :

- si deux sites se projettent orthogonalement sur l au même point, l'un de ces deux sites peut être supprimé pour résoudre le problème de l'intersection de $\mathcal{V}_{\mathcal{P}}$ avec l (première observation de BREU ET AL.) ;
- l'ordre des cellules sur l des sites de \mathcal{Q} est compatible avec l'ordre des projections des sites $\{p_i\}$ sur l , notées $\Pi_l(p_i)$ (seconde observation de BREU ET AL.).

De manière intuitive, ces deux propriétés sont presque triviales mais il apparaît qu'une preuve formelle de celles-ci n'est pas évidente.

Tout d'abord, considérons deux points A et B de \mathcal{P} , notons H l'hyper-plan médiateur du segment $[AB]$ et \vec{l} un vecteur directeur de l . Le premier lemme nous permet d'enlever de \mathcal{P} les points pathologiques, c'est-à-dire ceux dont les projections sur l sont confondues. Par la suite et pour simplifier les notations, on utilisera π_A pour $\Pi_l(A)$ et π_B pour $\Pi_l(B)$. Le lemme s'énonce alors de la façon suivante :

Lemme 4.1 *Le plan euclidien \vec{H} associé au plan affine H contient \vec{l} si et seulement si $\pi_A = \pi_B$*

Preuve : nous prouvons tout d'abord l'implication de gauche à droite. Nous supposons donc que $\vec{l} \in \vec{H}$, et notons M le milieu de $[AB]$ et $\Pi_l(M) = \pi_M$. Nous avons alors :

$$\vec{OA} = \vec{O\pi_M} + \vec{\pi_M M} + \vec{M\dot{A}}$$

Donc, si nous notons Π_l^* la projection vectorielle engendrée par Π_l , nous avons :

$$\begin{aligned} \Pi_l^*(\vec{OA}) &= \Pi_l^*(\vec{O\pi_M}) + \Pi_l^*(\vec{\pi_M M}) + \Pi_l^*(\vec{M\dot{A}}) \\ &= \vec{O\pi_M} + \vec{0} + \vec{0} \end{aligned}$$

Nous avons $\Pi_l^*(\vec{M\dot{A}}) = \vec{0}$ parce que \vec{AM} est orthogonal à tout vecteur appartenant à \vec{H} par définition, ainsi, comme $\vec{l} \in \vec{H}$, \vec{AM} est orthogonal à \vec{l} et donc $\Pi_l^*(\vec{M\dot{A}}) = \vec{0}$.

De plus, nous avons de la même manière :

$$\Pi_l^*(\vec{OA}) = \Pi_l^*(\vec{OB}) = \vec{O\pi_M}$$

Et donc finalement, $\pi_A = \pi_B = \pi_M$.

Pour prouver l'implication inverse, supposons $\pi_A = \pi_B$. Nous avons donc $\overrightarrow{\pi_M A} \perp \overrightarrow{l}$ et $\overrightarrow{\pi_M B} \perp \overrightarrow{l}$ par définition de la projection. Ainsi, \overrightarrow{l} est orthogonal au plan dans \mathbb{R}^n engendré par (π_A, A, B) . En particulier, $\overrightarrow{l} \perp \overrightarrow{AB}$ et comme \overrightarrow{H} contient tous les vecteurs orthogonaux à \overrightarrow{AB} , \overrightarrow{l} appartient nécessairement à \overrightarrow{H} . \square

Nous notons Σ_A (resp. Σ_B) l'intersection entre la cellule de Voronoï associée à A (resp. B) et la droite l (Σ_A et Σ_B peuvent être nuls). A partir du lemme précédent, nous avons les corollaires suivants :

Corollaire 4.1 *Si $\pi_A \neq \pi_B$, l'intersection entre H et l est réduite en un point et les deux segments sur l , Σ_A et Σ_B , sont disjoints.*

La preuve se déduit directement du lemme 4.1.

Corollaire 4.2 *Si $\pi_A = \pi_B$ nous avons deux cas possibles :*

- soit $\text{dist}(A, l) \neq \text{dist}(B, l)$ et donc si $\Sigma_A \neq \emptyset$ (resp. $\Sigma_B \neq \emptyset$) alors $\Sigma_B = \emptyset$ (resp. $\Sigma_A = \emptyset$), et donc $B \notin \mathcal{Q}$ (resp. $A \notin \mathcal{Q}$) ;
- soit $\text{dist}(A, l) = \text{dist}(B, l)$ et dans ce cas $\Sigma_A = \Sigma_B$ (les ensembles peuvent être éventuellement nuls) et donc soit A ou B (arbitrairement) peut ne plus être considéré comme appartenant à \mathcal{Q} .

Preuve : Le premier point de ce corollaire est direct, le second nécessite quelques détails. Montrons que $\Sigma_A = \Sigma_B$.

Soit p un point de Σ_A . Par définition, p appartient à l . Or, l est incluse dans H d'après le lemme 4.1. La cellule de Voronoï C_A engendrée par A est entièrement du même côté de H donc p est sur la frontière de C_A . Comme le diagramme de Voronoï est une partition de l'espace, il existe un site X tel que p soit sur la frontière de C_X . A et X sont des sites plus proches de p que tout autre site, Or nous avons $\text{dist}(A, p) = \text{dist}(B, p)$ donc soit $X = B$, soit p est aussi sur la frontière de B . Quel que soit le cas de figure, p appartient à Σ_B . Nous obtenons $\Sigma_A \subseteq \Sigma_B$.

De la même manière, nous pouvons prouver que $\Sigma_B \subseteq \Sigma_A$ et donc que $\Sigma_A = \Sigma_B$. \square

Maintenant, étant donné un tube dans une image discrète nD , nous avons le corollaire :

Corollaire 4.3 *Soit T un tube selon la k^{ime} dimension d'une image dans \mathbb{Z}^n , soit \mathcal{Q} la liste des sites de $p_i \in P$ (dans ce cas $P \subset \mathbb{Z}^n$) telle que $\mathcal{C}_{p_i} \cap T \neq \emptyset$, nous avons alors :*

$$|\mathcal{Q}| \leq m_k \tag{4.9}$$

D'après le corollaire 4.2 et dans une optique de transformée en distance, nous pouvons considérer uniquement des sites de \mathcal{P} tels que $\pi_A \neq \pi_B$ (pour tout couple (A, B)).

Nous présentons maintenant le lemme montrant la préservation de l'ordre entre les cellules et la projection de sites sur l :

Lemme 4.2 *Supposons $\mathcal{P} = \{A, B\}$, $\pi_A \neq \pi_B$, $a \in \Sigma_A$ et $b \in \Sigma_B$ ($a \neq b$), l'ordre des points (π_A, π_B) sur l est le même que (a, b) , ce qui s'écrit :*

$$\vec{ab} = \kappa \overrightarrow{\pi_A \pi_B} \quad (\text{avec } \kappa > 0) \quad (4.10)$$

où $\cdot \cdot$ est le produit scalaire de vecteurs.

Preuve : Par définition, A et B sont de part et d'autre de l'hyperplan H et \overrightarrow{AB} est normal au plan. Or, a est du même coté de H que A , de même pour b qui est du même coté de H que B . Donc la composante suivant \overrightarrow{AB} du vecteur \vec{ab} est positive. En d'autres termes :

$$\vec{ab} \cdot \overrightarrow{AB} = \lambda \quad (\lambda > 0) \quad (4.11)$$

Par définition de la projection, la composante suivant $\overrightarrow{\pi_A \pi_B}$ du vecteur \overrightarrow{AB} est $\overrightarrow{\pi_A \pi_B}$. Ainsi :

$$\overrightarrow{\pi_A \pi_B} \cdot \overrightarrow{AB} = \|\overrightarrow{\pi_A \pi_B}\|^2 = \mu \quad (\mu > 0) \quad (4.12)$$

En remplaçant \vec{ab} par sa valeur (4.10) dans l'équation (4.11), nous obtenons

$$\kappa \overrightarrow{\pi_A \pi_B} \cdot \overrightarrow{AB} = \lambda$$

C'est-à-dire :

$$\kappa = \frac{\lambda}{\mu} > 0$$

□

Nous pouvons maintenant conclure sur tous les points de \mathcal{P} en appliquant le lemme précédent sur chaque paire de points de \mathcal{P}

Corollaire 4.4 *Si les sites $\{p_i\}_{1..n}$ de \mathcal{P} sont tels que $\Pi_l(p_1) < \dots < \Pi_l(p_n)$, alors l'ordre des Σ_{p_i} sur l est compatible.*

Preuve : si on considère l'intersection de $\mathcal{V}_{\mathcal{P}}$, pour chaque paire de points (p, q) de \mathcal{P} tels que $\Pi_l(p) \neq \Pi_l(q)$, les cellules, sur l , Σ_p et Σ_q sont des sous-ensembles (peut-être vides) des cellules du diagramme de Voronoï qui ne considère que les points p et q . Ainsi, d'après le lemme 4.2, l'ordre sur l des projetés $\Pi_l(p)$ et $\Pi_l(q)$ est le même que celui des cellules Σ_p et Σ_q . Étant donné que cette propriété est vraie pour tout couple de points de \mathcal{P} , elle est vraie pour tous les points de \mathcal{P} . □

Sous l'hypothèse que $\Pi_l(u) < \Pi_l(v) < \Pi_l(w)$, nous pouvons construire un prédicat `CACHÉ_PAR_l` qui permet de décider si v est caché par u et w sur la droite l et ce quelle que soit la dimension.

Considérons les sites $\{p_i\}$ de \mathcal{P} tels que :

$$\Pi_l(p_i) < \Pi_l(p_j) \quad \forall (i, j) \ 1 \leq i < j \leq n \quad (4.13)$$

Ainsi, en utilisant le prédicat `CACHÉ_PAR_l` et l'algorithme 17 de suppression de sites, nous avons le corollaire suivant :

Corollaire 4.5 *Étant donné un ensemble de site \mathcal{P} vérifiant l'équation (4.13) et un prédicat `CACHÉ_PAR_l`, nous calculons l'ensemble des sites $\{q_i\}$ de \mathcal{Q} tels que $\Sigma_{q_i} \neq \emptyset$ en $O(|\mathcal{P}|)$ avec l'algorithme 17.*

4.2.6.2 Description de l'algorithme

Étant donné que toutes les observations de BREU ET AL. ont été montrées en dimension quelconque dans le paragraphe précédent, nous pouvons proposer un algorithme de calcul de diagramme de Voronoï discret et de transformée en distance euclidienne (voir algorithme 21).

Algorithme 21 Diagramme de Voronoï discret en nD

DIAGRAMME_VORONOÏ_DISCRET_ND(I,P)

- 1: Calcul des diagrammes de Voronoï discrets uni-dimensionnels comme initialisation
 - 2: **pour tout** dimension $k \geq 2$ **faire**
 - 3: **pour tout** tube T selon la dimension k **faire**
 - 4: Construction du prédicat CACHÉ_PAR_T
 - 5: $Candidates = \{\text{sites des diagrammes de Voronoï discrets dans les } (k - 1) \text{ dimensions le long du tube } T\}$
 - 6: Construction de la liste \mathcal{L}_T à partir de $Candidates$ en utilisant CACHÉ_PAR_T et l'algorithme 17
 - 7: **fin du pour**
 - 8: **fin du pour**
-

Théorème 4.1 *L'algorithme 21 est correct et sa complexité est linéaire en le nombre de points de l'image.*

Preuve : *la preuve de la correction de cet algorithme est similaire à celle proposée par SAITO et TORIWAKI (1994) pour une approche basée sur des tubes le long des différentes dimensions. De plus, les preuves de la section 4.2.6.1 attestent la correction du processus de suppression de sites pour le cas nD à toute étape k de l'algorithme.*

Concernant la preuve de linéarité, nous utilisons le corollaire 4.3 et le fait que l'algorithme 17 est linéaire en la taille de la liste d'entrée (le coût du prédicat CACHÉ_PAR_T est toujours $O(1)$ quelle que soit la dimension). Ainsi, pour chaque dimension k de 1 à n , le calcul des diagrammes de Voronoï discrets partiels se font en temps linéaire (en le nombre de points de l'image).

Si les dimensions de l'image binaire sont données par $[1, m_1] \times \dots \times [1, m_n]$, l'algorithme précédent a un coût en $O(m_1 m_2 \dots m_n)$ ce qui est optimal pour le problème. \square

4.2.7 Analyse expérimentale

Dans un premier temps, nous illustrons la transformée en distance et le diagramme de Voronoï discret en dimension 2. Il est en effet plus simple d'observer les structures dans cette dimension qu'en dimension 3 directement.

Considérons tout d'abord une distribution aléatoire de sites (voir figure 4.10). Nous représentons donc le diagramme de Voronoï discret en affectant un niveau de gris pour chaque cellule. La figure 4.10-(c) présente la transformée en distance euclidienne exacte et pour des raisons de lisibilité, nous utiliserons par la suite une palette de niveaux de gris réduite pour représenter cette transformée (figure 4.10-(d)).

Dans la figure 4.11, nous calculons le diagramme de Voronoï ainsi que la transformée en distance d'un disque. Pour être plus précis, les sites considérés dans ce cas

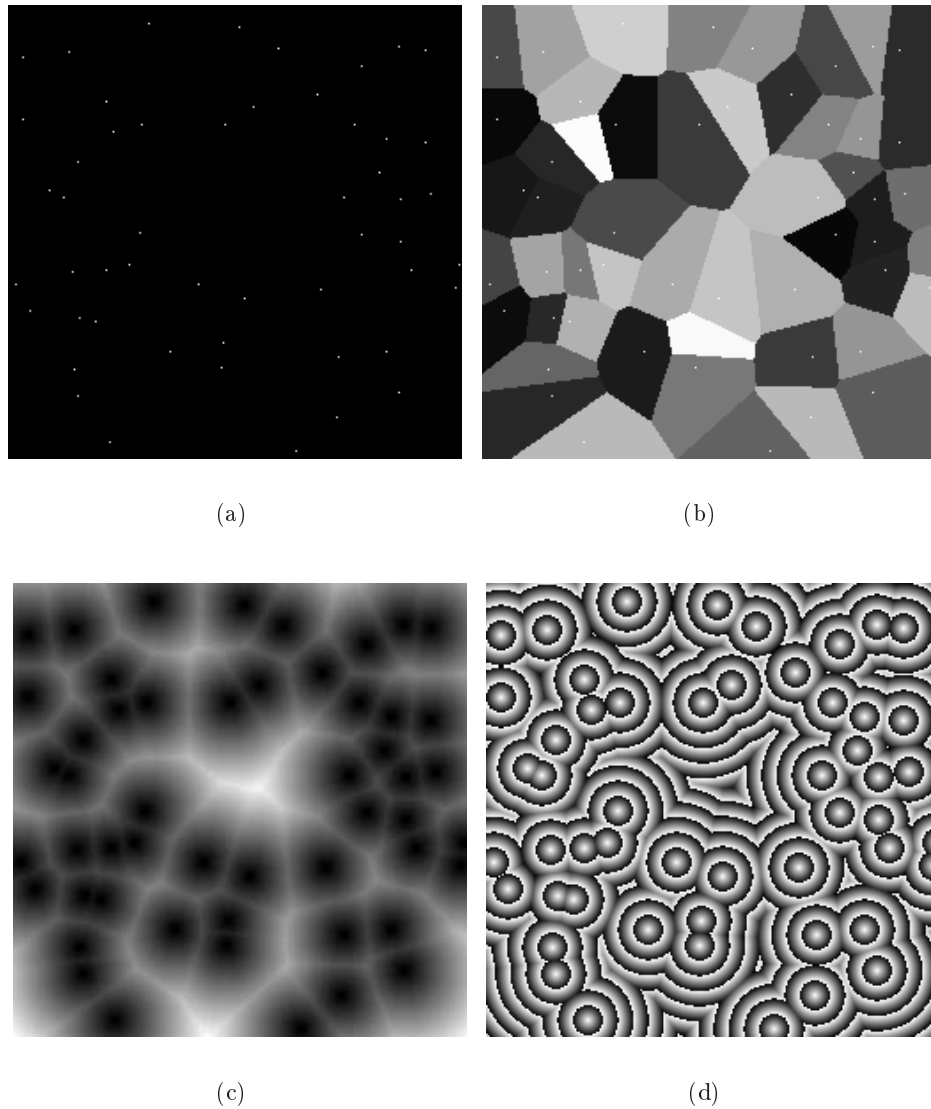


FIG. 4.10 —: *Diagramme de Voronoï et EDT de sites distribués aléatoirement : (a) image binaire des sites, (b) diagramme de Voronoï dans lequel un niveau de gris est affecté à chaque site, (c) Transformée en distance euclidienne (histogramme normalisé) et (d) EDT avec une palette de niveaux de gris réduite pour mieux visualiser les fronts d'ondes.*

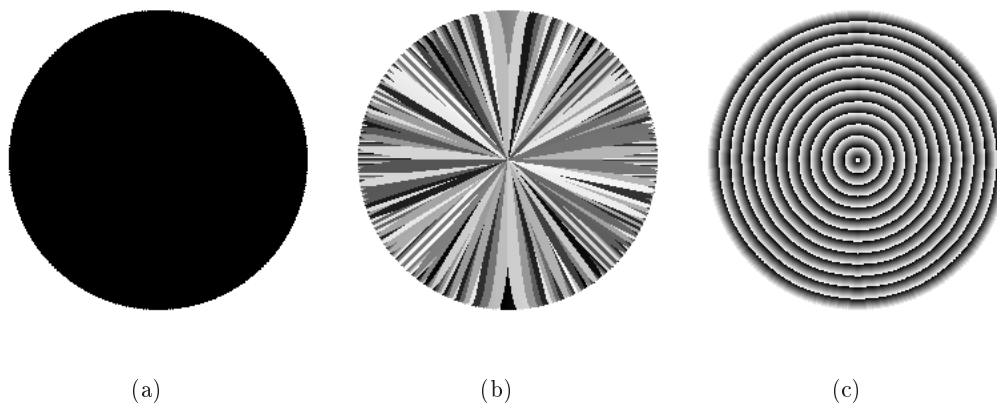


FIG. 4.11 –: *Diagramme de Voronoï et EDT d'un disque.*

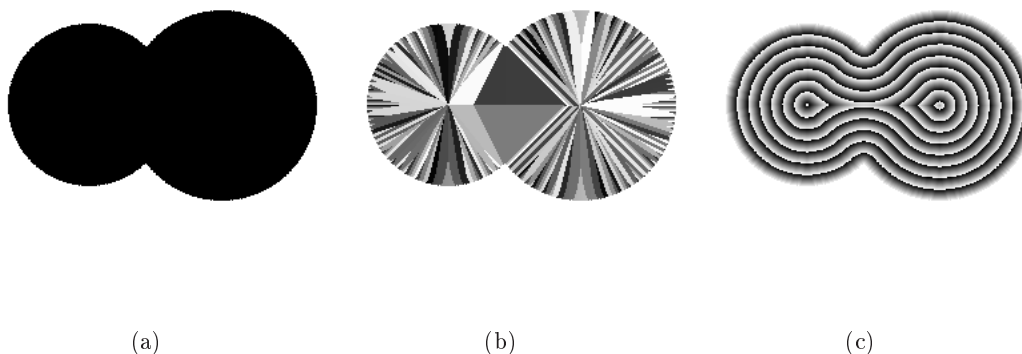


FIG. 4.12 –: *Diagramme de Voronoï et EDT de deux disques avec une intersection non vide : le disque de droite est légèrement plus grand que celui de gauche ce qui permet d'expliquer le non respect de la symétrie.*

sont les pixels complémentaires de l'objet. Dans ce cas, nous pouvons voir très nettement le faisceau des cellules du diagramme de Voronoï qui *convergent* vers le centre du disque. La figure 4.12 considère une forme composée de deux disques joints.

La figure 4.13 illustre les algorithmes sur un rectangle plein. Dans l'illustration du diagramme de Voronoï, les phénomènes de dégradé sont dus à l'algorithme d'affectation d'un niveau de gris pour chaque cellule. Cependant, nous pouvons remarquer *le squelette* de la forme sur le diagramme que nous formaliserons par la suite.

Enfin, la figure 4.14 présente les différents algorithmes sur toute une variété d'objets synthétiques.

Pour illustrer les différents algorithmes en 3D, nous considérons tout d'abord l'image binaire de la figure 4.15-(a) qui se compose d'un cube discret, les sites sont placés à chaque sommet de celui-ci ainsi qu'à son centre. Les figures 4.15-(b) à 4.15-(d) représentent les différents diagrammes de Voronoï discrets partiels après respectivement la première phase d'initialisation, la fusion en y puis la fusion en z . Les figures 4.15-(e) à 4.15-(g) correspondent à des visualisations du diagramme de

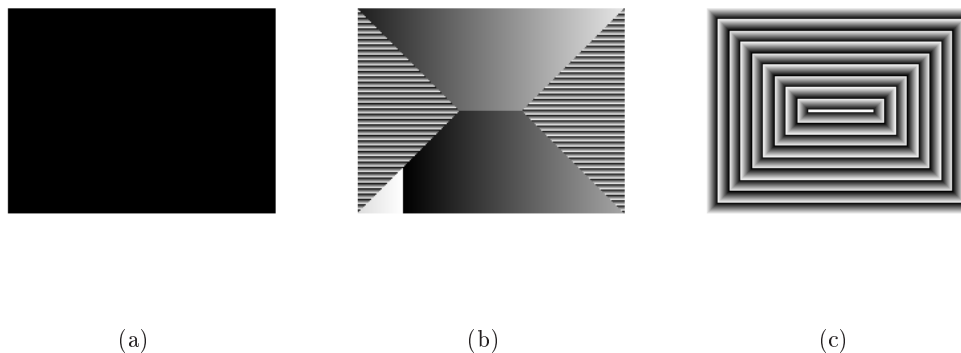


FIG. 4.13 –: *Diagramme de Voronoï et EDT d'un rectangle : (a) image source, (b) Diagramme de Voronoï et (c) EDT avec une palette de niveaux de gris réduite.*

Voronoi 3D suivant différents plans de coupe.

Dans la figure 4.16, nous illustrons le diagramme de Voronoï discret 3D et la transformée en distance euclidienne d'une grille ne contenant que trois sites.

Dans le cadre d'analyse de formes 3D, la figure 4.17 présente un calcul de transformée en distance euclidienne sur un objet synthétique. Nous utilisons dans ce schéma un plan de coupe afin de voir ce qu'il se passe à l'intérieur de la forme. Pour cette image, nous pouvons observer les bons comportements de la transformée aussi bien sur les parties sphériques de la forme que sur la partie parallélépipédique.

Finalement, nous pouvons aussi considérer la transformation inverse qui prendrait en entrée un squelette et qui reconstruirait la surface. La figure 4.18 illustre ce processus de reconstruction par surface implicite de l'objet à partir d'un squelette.

4.2.8 Diagramme de Voronoï discret et transformée en distance sur des grilles non-cubiques

Dans notre approche, nous introduisons une structure géométrique complexe, le diagramme de Voronoï, entre l'objet binaire et sa transformée en distance. Cette étape supplémentaire nous permet d'étendre ces algorithmes pour des grilles non-cubiques.

Dans un premier temps, nous considérons des grilles parallélépipédiques. En pratique, beaucoup d'appareils d'acquisition classiques en imagerie médicale produisent des grilles qui ne sont pas isotropes. Par exemple, pour certains types d'appareils et pour certains types de partie du corps, les rapports entre les directions peuvent aller jusqu'à 3 ou 5.

De nombreux travaux existent pour résoudre le problème de la transformée en distance sur ce type de grille. Dans le cas des distances de chanfrein, COQUIN ET AL. (1994) et SINTORN et BORGEFORS (2002) proposent, par exemple, un calcul des poids du masque adapté en fonction de facteurs d'échelle entre les axes. Dans ce cas, si ce facteur d'échelle varie entre deux images volumiques, il faut recalculer les poids optimaux. Pour le cas de la transformée en distance, l'algorithme de SAITO et TORIWAKI (1994), optimisé par HIRATA (1996) et MEIJSTER ET AL. (2000), permet de calculer la transformée exacte et résout ainsi le problème. Nous montrons, par la suite, comment notre approche permet elle aussi de résoudre ce problème avant de

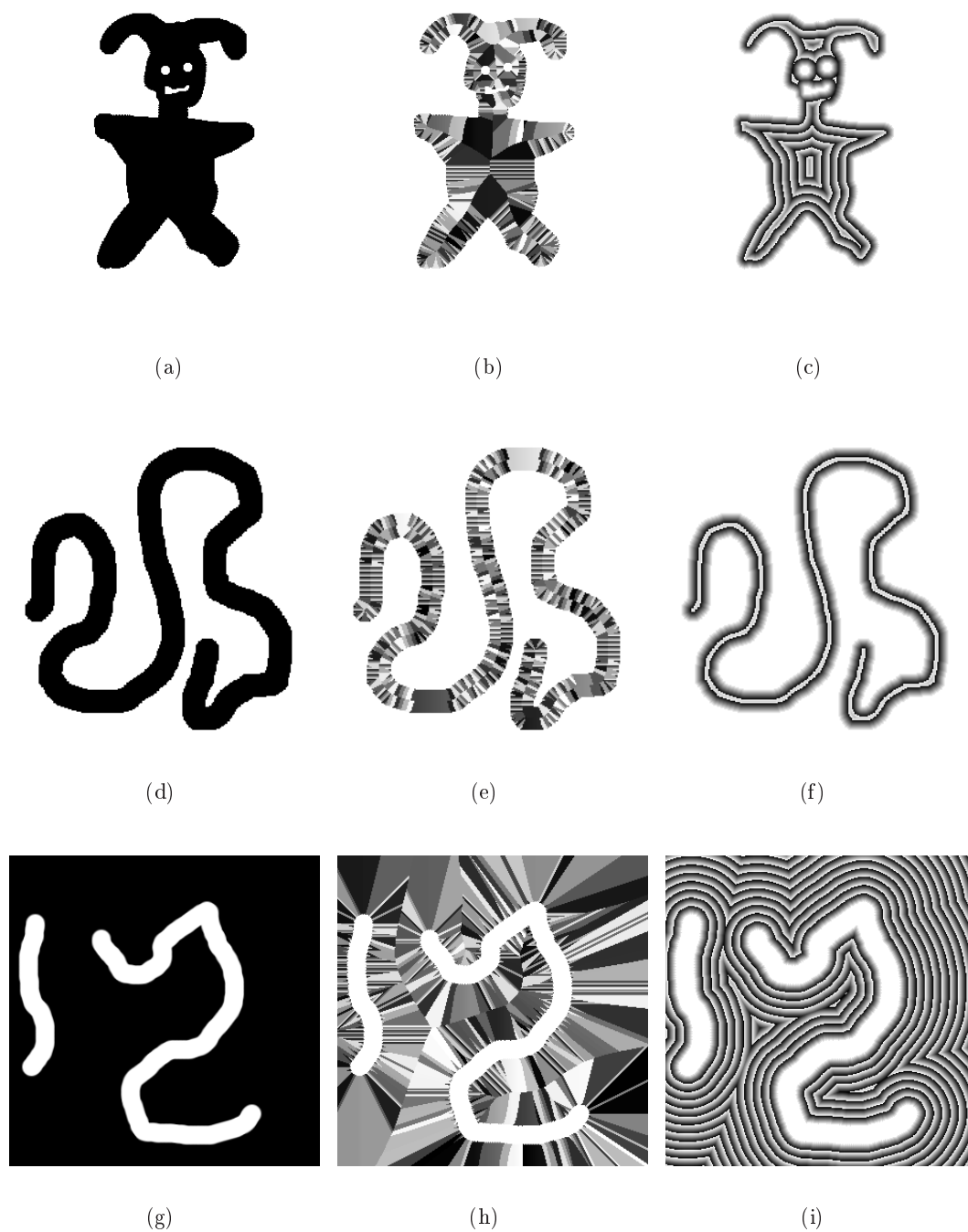


FIG. 4.14 —: *Diagramme de Voronoï et EDT d'objets synthétiques.*

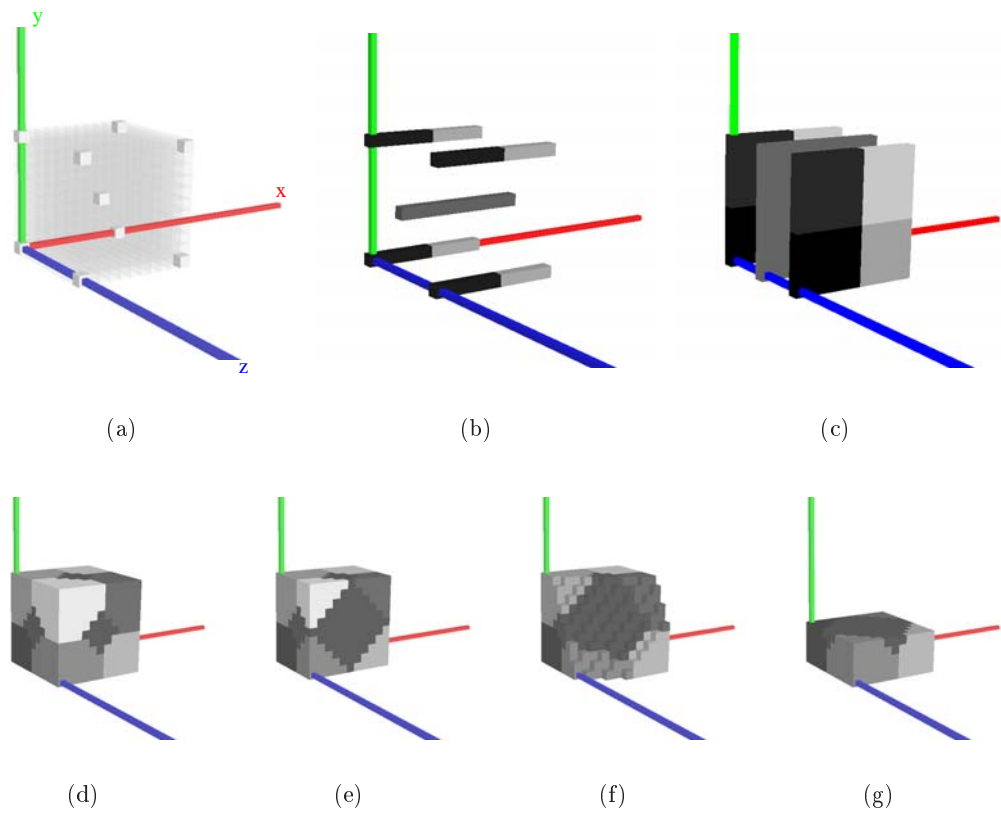


FIG. 4.15 —: *Illustration du diagramme de Voronoï discret 3D : (a) l'image binaire avec les sites en gris, (b) diagrammes de Voronoï discrets uni-dimensionnels après la première phase d'initialisation selon l'axe Ox , (c) diagrammes de Voronoï discrets 2D après la fusion en y , (d) diagramme de Voronoï discret 3D et (e) – (g) le diagramme 3D avec des plans de coupe pour voir les structures internes.*

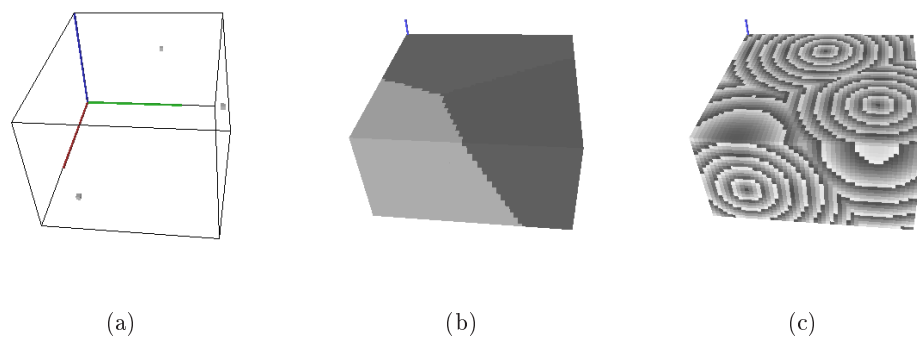


FIG. 4.16 —: *Illustration du diagramme de Voronoï discret 3D et de la transformée en distance : (a) image source avec trois sites, (b) diagramme de Voronoï discret 3D et (c) la transformée en distance (avec une palette réduite).*

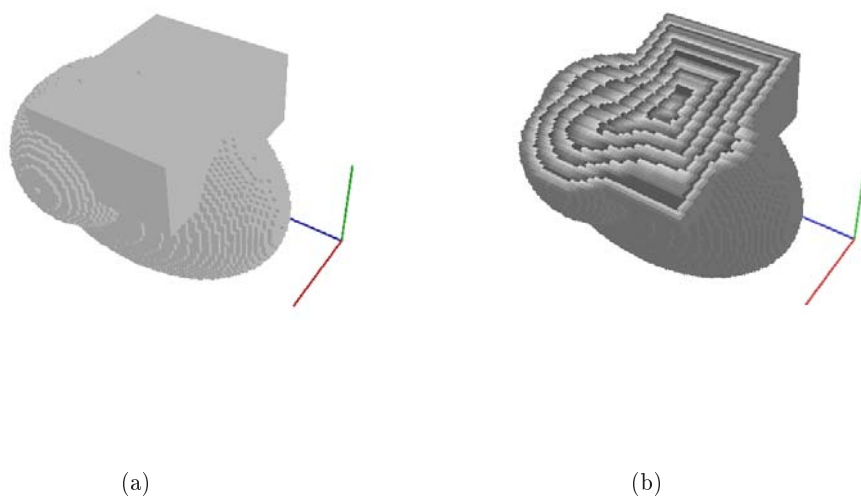


FIG. 4.17 –: Transformée en distance euclidienne 3D : (a) image source et (b) transformée en distance avec une palette réduite et un plan de coupe.

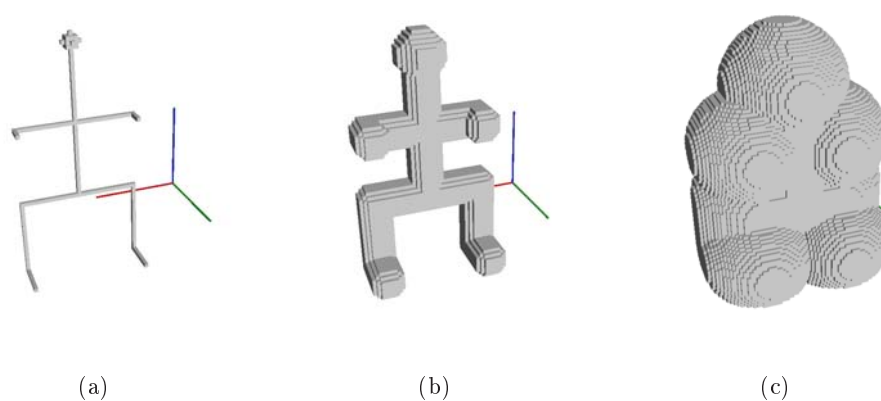


FIG. 4.18 –: Reconstruction par surface implicite à partir d'un squelette : (a) le squelette pouvant être vu comme un ensemble de sites, (b) reconstruction d'une iso-surface à une distance 5 du squelette et (c) reconstruction à une distance 20.

passer à des espaces discrets plus complexes (grille hexagonale par exemple).

Soit \mathcal{P} un ensemble de sites tel que $\mathcal{P} = \{(x, y, z) \in \mathbb{Z}^3\}$. Nous notons $\mathcal{P}_{\alpha\beta}$ l'ensemble de sites $(x, \alpha y, \beta z)$ avec $(x, y, z) \in \mathbb{Z}^3$ et $\alpha, \beta \in \mathbb{R}^2$

Proposition 4.2 *Soit $\mathcal{V}_{\mathcal{P}}$ le diagramme de Voronoï associé aux sites \mathcal{P} et $\mathcal{V}_{\mathcal{P}_{\alpha\beta}}$ le diagramme de Voronoï associé aux sites $\mathcal{P}_{\alpha\beta}$, alors :*

$$\mathcal{V}_{\mathcal{P}_{\alpha\beta}} = \mathcal{E}_{\alpha\beta}(\mathcal{V}_{\mathcal{P}})$$

où $\mathcal{E}_{\alpha\beta}$ correspond à un changement d'échelle de α suivant y et un changement d'échelle de β suivant z .

Preuve : *La preuve est directe, on considère des points p, q , et r de \mathcal{P} et p', q' et r' leurs analogues³ dans $\mathcal{P}_{\alpha\beta}$, alors nous avons :*

$$\text{dist}(p, q) < \text{dist}(p, r) \Leftrightarrow \text{dist}(p', q') < \text{dist}(p', r')$$

où dist est la distance euclidienne.

Ainsi, si un point p appartient à la cellule du site q dans $\mathcal{V}_{\mathcal{P}}$, alors il appartient à la cellule du site q' dans $\mathcal{V}_{\mathcal{P}_{\alpha\beta}}$, et réciproquement. \square

Finalement, nous pouvons utiliser exactement les mêmes algorithmes pour extraire le diagramme de Voronoï discret de la grille \mathbb{Z}^3 . Il suffit de les transformer avec les changements d'échelle $\mathbb{Z} \times \alpha\mathbb{Z} \times \beta\mathbb{Z}$ et ainsi obtenir, après réécriture, la transformée en distance euclidienne sur ces grilles (voir figure 4.19). Bien évidemment, ce que nous avons illustré en 3D est aussi valable quelle que soit la dimension et donc les algorithmes proposés s'adaptent aussi pour des grilles anisotropiques en dimension n .

Si nous considérons maintenant des grilles non-cubiques, par exemple des grilles hexagonales, très peu de solutions existent pour le calcul de la transformée en distance. Cependant, dans le cas particulier des grilles hexagonales, de nombreux objets à analyser existent, notamment en physique.

Tout d'abord, nous pouvons remarquer qu'à aucun endroit de la méthode ni de son analyse théorique le fait que les tubes soient parallèles aux axes de la grille n'est utilisé. En d'autres termes, nous pouvons très bien envisager des suppressions de sites, dans des tubes orthogonaux, qui ne soient pas forcément parallèles aux axes. En effet, nous pouvons effectuer le processus de fusion avec un tube donné par une équation du type $y = \alpha x + \beta$, il suffit dans ce cas d'adapter le prédicat `CACHÉ_PAR` pour tenir compte de l'équation. Considérons pour cela une droite réelle notée l et telle que $y = \alpha x + \beta$, ainsi que deux points u et v tels que $\Pi_l(u) \neq \Pi_l(v)$ ce qui s'écrit dans notre exemple $2(u_x - v_x) + 2\alpha(v_y - u_y) \neq 0$. Dans ce cas nous avons les coordonnées du point $\hat{u}v$, intersection de la médiatrice de $[uv]$ et de l , qui sont données par :

$$\hat{u}v = \left(\frac{u_x^2 + u_y^2 - v_x^2 - v_y^2 + 2\beta(v_y - u_y)}{2(u_x - v_x) + 2\alpha(v_y - u_y)}, \frac{\alpha(v_x^2 + v_y^2 - u_x^2 - u_y^2) + 2\beta(u_x - v_x)}{2(u_x - v_x) + 2\alpha(v_y - u_y)} \right)$$

Ainsi, il est facile de construire un prédicat `CACHÉ_PAR`(α, β) qui permet de tester, étant donnés trois sites u, v et w tels que $\Pi_l(u)_y < \Pi_l(v)_y < \Pi_l(w)_y$ (on

³. un point (x, y, z) de \mathcal{P} a pour analogue dans $\mathcal{P}_{\alpha\beta}$ le point $(x, y/\alpha, z/\beta)$.

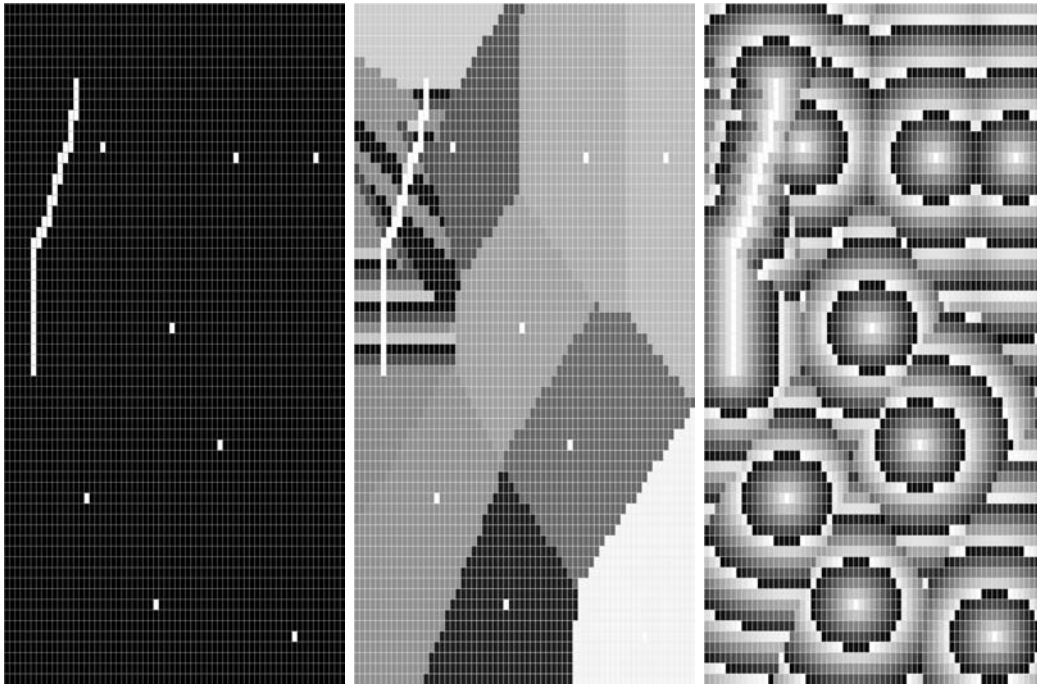


FIG. 4.19 – : *Diagramme de Voronoï et EDT sur une grille anisotrope : le facteur d'échelle entre l'axe Ox et l'axe Oy est de 2.*

suppose $\alpha \neq 0$), si la cellule associée à v est *cachée* par les cellules de u et de w sur la droite l . En ne changeant que ce simple prédicat et la construction des tubes (qui peut être basée sur les notions de droites discrètes présentées dans le chapitre 2.2), nous réutilisons tous les algorithmes présentés pour le diagramme de Voronoï discret et la transformée en distance euclidienne.

Bien évidemment, utiliser des tubes qui ne sont pas parallèles aux axes n'a pas beaucoup d'intérêt dans le cas de grilles orthogonales. Cependant, dans le cas de grilles hexagonales, cette propriété est fondamentale. Considérons l'échantillonnage de la grille proposé dans la figure 4.20 qui consiste à ajouter aux points discrets du maillage certains points qui sont sur les arêtes *en bas à gauche*. Ainsi, en utilisant une phase d'initialisation dans les tubes paramétrés par $y = \sqrt{3}x + b$ puis en appliquant le processus de suppression de sites avec le prédicat `CACHÉ_PAR_`($-1/\sqrt{3}, b'$) dans les tubes orthogonaux aux précédents, nous obtenons un algorithme de calcul du diagramme de Voronoï discret sur notre maillage. Étant donné que l'échantillonnage est linéaire en le nombre de points de la grille hexagonale d'origine, la complexité du calcul reste linéaire en le nombre de points et donc toujours optimal pour le problème. La figure 4.21 illustre ce calcul.

Si maintenant nous revenons à notre grille initiale d'hexagones, nous construisons le diagramme de Voronoï discret pour cette grille en ne conservant du diagramme que les points qui sont au centre des hexagones. Nous obtenons ainsi le diagramme de Voronoï discret et la transformée en distance euclidienne en temps optimal pour ce type de grille.

4.2.9 Conclusion et perspectives

Nous avons montré dans cette partie un algorithme calculant la transformée en distance euclidienne exacte sur des images de dimension n en temps optimal, c'est-à-dire

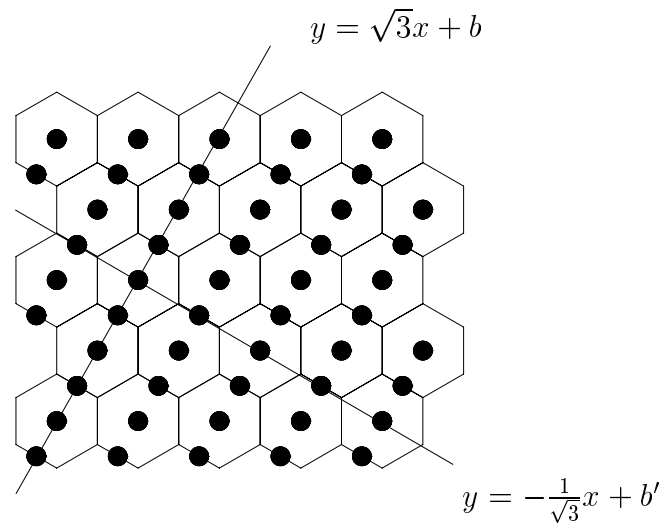


FIG. 4.20 –: Rééchantillonnage de la grille hexagonale pour le calcul du diagramme de Voronoï discret

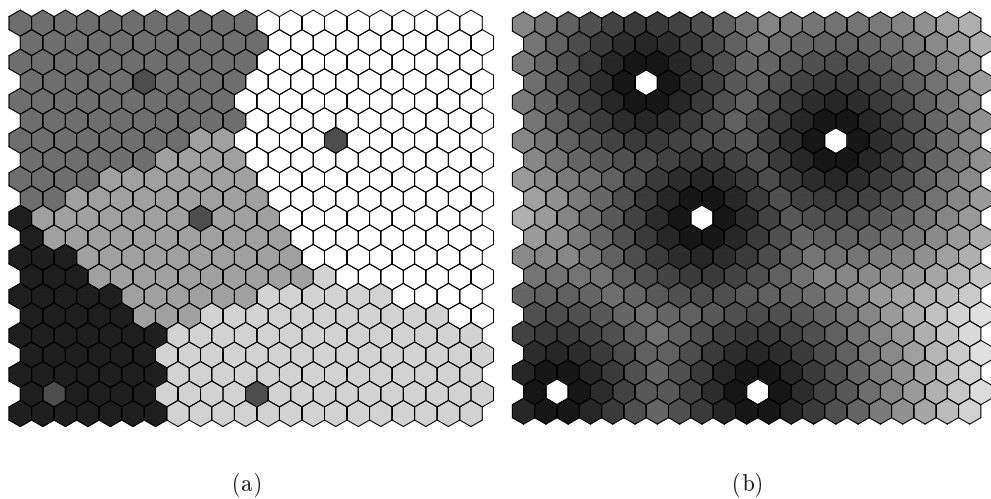


FIG. 4.21 –: Diagramme de Voronoï et EDT sur une grille hexagonale : (a) diagramme de Voronoï discret (les sites sont les cellules noires) et (b) transformée en distance euclidienne.

en temps linéaire en le nombre de points discrets de la grille. Cet algorithme se base sur la construction d'une étape intermédiaire qui est diagramme de Voronoï discret. Nous avons montré l'intérêt de cette construction dans le cas des grilles anisotropiques et des grille hexagonales.

4.3 Squelette de forme et axe médian

Maintenant que nous avons des algorithmes optimaux pour le calcul de la transformée en distance, nous allons extraire de celle-ci un objet géométrique appelé *squelette* ou *axe médian* qui permet une description géométrique et topologique de la forme. Nous présentons donc les approches classiques pour construire de tels objets en géométrie discrète puis nous proposons un algorithme direct et optimal en temps basé sur le diagramme de Voronoï discret que nous venons de définir.

4.3.1 Axe médian et squelette en géométrie discrète

Comme nous l'avons déjà présenté dans le paragraphe 4.2, la notion d'axe médian proposée par BLUM (1967) correspond à une généralisation des axes de symétrie d'une forme. ROSENFELD et PFALTZ (1966) parlent de squelette de forme qui est défini comme étant le sous-ensemble minimal de la forme permettant la reconstruction de celle-ci en appliquant une transformée en distance inverse.

En géométrie discrète, nous trouvons généralement le terme *squelette* pour décrire un objet topologique homotope à la forme (nous reviendrons sur ce point plus tard). Dans ce qui suit et quand nous ne précisons pas, nous parlerons toujours de squelette géométrique.

Nous pouvons définir formellement cette notion de *minimalité* en introduisant la notion de *boule maximale* pour une métrique donnée.

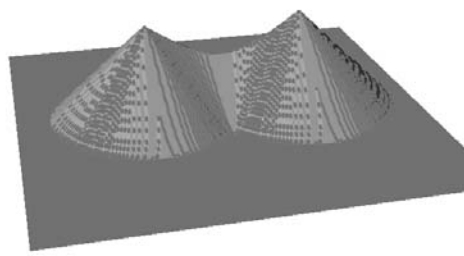
Définition 4.4 (Boule maximale) *Une boule, au sens de la métrique considérée, de centre p et de rayon r est dite maximale si celle-ci est entièrement contenue dans la forme et qu'il n'existe aucune autre boule de la forme qui la contienne.*

Ainsi, l'axe médian se définit de la façon suivante :

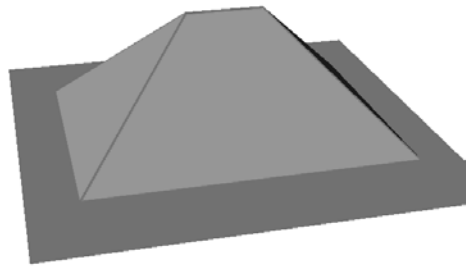
Définition 4.5 (Axe médian) *L'axe médian d'une forme est l'ensemble des centres des boules maximales de celle-ci.*

Selon ces définitions, on peut tout de suite remarquer que le test qui permet de décider si un point p est centre de boule maximale est primordial. Pour un point p , le rayon de la boule susceptible d'être maximale est bien évidemment donné par la transformée en distance selon la métrique considérée. La figure 4.22 illustre cette notion de maximum local dans la carte de distance : la transformée en distance est représentée sous forme de surface dont la hauteur est la valeur de la transformée. Sur ce type de schéma, une définition intuitive de l'axe médian correspond aux *crêtes* des ces surfaces.

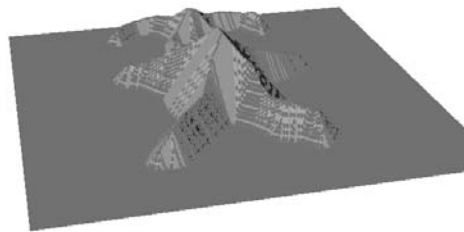
Pour calculer cet axe médian, plusieurs approches sont envisageables. Nous présenterons uniquement une approche récente de REMY (2001). Étant donné un point de la transformée en distance, décider si celui-ci est centre de boule maximale consiste à regarder dans un certain voisinage s'il n'existe pas de centres de boules plus grosses



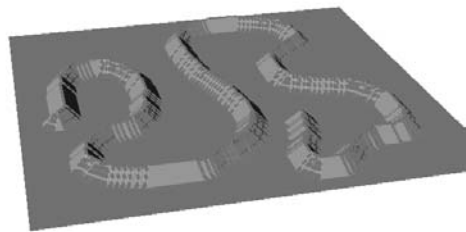
(a)



(b)



(c)



(d)

FIG. 4.22 —: *Représentation surfacique de la transformée en distance : l'axe médian correspond aux crêtes de ces surfaces.*

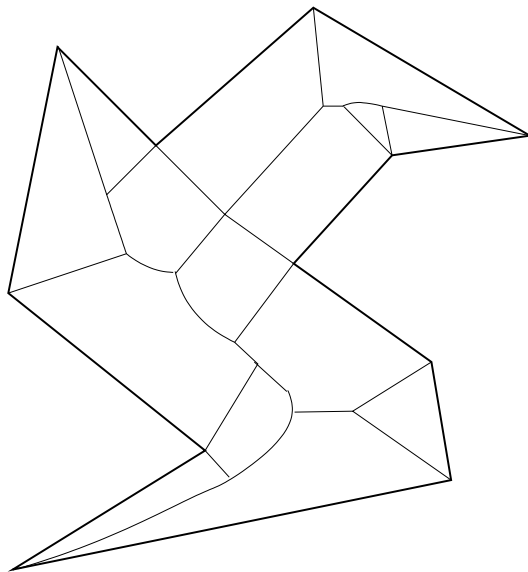


FIG. 4.23 –: *Squelette d'un polygone basé sur le diagramme de Voronoï généralisé (extrait de ATTALI 1995)*

contenant la première. De manière plus formelle, on considère un point p de la transformée à une distance d du bord et un point q à une distance d' du bord. L'idée est la suivante : quelles sont les conditions sur d' pour que la boule $B(q, d')$ contienne la boule $B(p, d)$? Pour résoudre ce problème on peut pré-calculer les valeurs de d' pour, étant donné le déplacement \vec{pq} et la valeur d , décider si p fait parti de l'axe médian. Ainsi, le test d'appartenance à l'axe médian s'effectue après une recherche dans une table de correspondance pré-calculée (Look-Up Table ou LUT dans la bibliographie). REMY (2001) propose donc une méthodologie générale pour le calcul de cette table de correspondance pour des métriques de chanfrein et le carré de la distance euclidienne.

Il existe d'autres types de squelettes d'objets discrets, basé par exemple sur des propriétés uniquement topologiques. Ainsi, le squelette *topologique* d'une forme peut se définir comme étant le sous-ensemble minimal de points discrets de l'objet ayant la même homotopie que l'objet global (même nombre de trous, même nombre de composantes connexes...). Ce genre de squelette se construit par exemple avec des suppressions séquentielles ou parallèles de points de la forme vérifiant un certain nombre de critères locaux (notion de *point simple*) mais garantissant une propriété globale d'homotopie (voir par exemple LOHOU et BERTRAND 2002).

Nous ne nous intéresserons par la suite qu'aux squelettes *géométriques*

4.3.2 Squelette n -dimensionnel basé sur le diagramme de Voronoï discret

Considérons dans un premier temps une forme polygonale. Dans ce cas, le squelette et le diagramme de Voronoï généralisé intérieur de la forme coïncident (voir figure 4.23). En effet, les sommets de ce diagramme ainsi que les arêtes sont centres de boules maximales pour la forme. Une des premières approches pour construire ce squelette a été proposée par MONTANARI (1969).

Dans le cas de formes quelconques, ATTALI (1995) propose un calcul de squelette

en considérant le diagramme de Voronoï interne d'un échantillonnage du bord de la forme. Ce squelette approxime le squelette exact de la forme continue sous-jacente mais a des propriétés de convergence si l'on considère un échantillonnage tendant vers l'infini.

De manière plus précise, étant donné un échantillonnage et le diagramme de Voronoï de celui-ci, le squelette peut être extrait du diagramme selon plusieurs approches (voir ATTALI 1995) :

- ensemble des sommets du diagramme inclus dans la forme
- ensemble des éléments du diagramme (sommets et arêtes) inclus dans la forme
- intersection entre le diagramme et la forme
- dual de la forme

Dans les algorithmes présentés précédemment, non seulement nous avons une transformée en distance sans erreur en temps linéaire en le nombre de points de la forme, mais nous avons surtout dans le même temps le diagramme de Voronoï discret de l'ensemble des points du complémentaire. Il est donc tout à fait normal de construire un squelette directement sur ce diagramme plutôt que sur la transformée en distance. L'information primordiale pour une extraction de squelette que n'ont pas les autres algorithmes de transformée en distance, c'est le lien entre tout point de la forme et le point du complémentaire le plus proche.

Nous définissons donc notre squelette de la manière suivante :

Définition 4.6 (Squelette basé sur diagramme de Voronoï discret) *Étant donnée une forme discrète et le diagramme de Voronoï discret de son complémentaire, le squelette associé à cette forme est l'ensemble des points discrets sommets ou sur les arêtes du diagramme de Voronoï discret.*

Comme le souligne ATTALI, un squelette défini de la sorte ne peut pas être utilisé directement, il faut le simplifier. Une des difficultés introduites par le diagramme discret est que nous ne connaissons pas la structure globale du graphe du diagramme de Voronoï, nous savons uniquement pour tout point, la cellule à laquelle il appartient. Ainsi, nous ne pouvons utiliser que des critères locaux pour la simplification du squelette. Nous considérons donc les filtrages locaux du squelette proposés par ATTALI. Pour cela, nous reprenons les définitions suivantes (voir figure 4.24) :

Définition 4.7 (Fonction d'étanchéité (ATTALI, 1995)) *Soit X une forme continue et $Sk(X)$ son squelette. La fonction d'étanchéité $\rho(\cdot)$ est la fonction qui à chaque point s du squelette associe le rayon de la boule maximale centrée en s .*

Cette fonction correspond à la donnée de la transformée en distance en tout point du squelette.

Définition 4.8 (Angle bissecteur (ATTALI, 1995)) *Soit X une forme continue et $Sk(X)$ son squelette. Soit s un point du squelette. La boule maximale centrée en s possède deux points de contact p_0 et p_1 avec la frontière de X . On appelle angle bissecteur l'angle $\widehat{p_0sp_1}$ compris entre 0 et π . Par la suite, on note $\alpha(s)$ l'angle bissecteur au point s . L'angle bissecteur aux points terminaux et multiples se calcule par passage à la limite.*

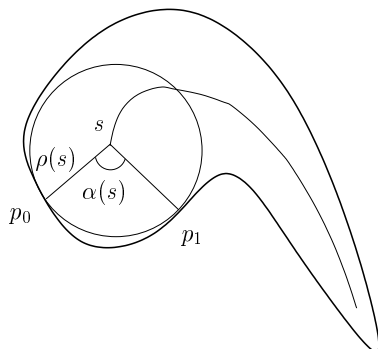


FIG. 4.24 –: *Illustration de la fonction d’étanchéité et l’angle bissecteur de ATTALI (1995).*

De manière informelle, on ne voudrait garder comme points du squelette que des points qui sont centres de boules maximales *assez grosses* et qui ne sont pas des extrémités *trop vives* du squelette. Nous filtrons donc les points du squelette construits précédemment avec deux seuils α_0 et ρ_0 et nous gardons les points dont l’angle bissecteur $\alpha(s)$ et l’étanchéité $\rho(s)$ vérifient :

$$\alpha(s) > \alpha_0 \quad \text{et} \quad \rho(s) > \rho_0$$

Nous appliquons cet algorithme d’extraction de squelette sur des formes géométriques les plus diverses. La figure 4.25, représente le calcul du squelette d’un disque. Pour illustrer le processus de simplification du squelette, nous représentons tous les points de celui-ci sur un graphe dans lequel en abscisse se trouve la valeur de l’angle bissecteur et en ordonnée la valeur de la fonction d’étanchéité. Ainsi, un filtrage correspond à extraire les points de ce graphe contenus dans une certaine fenêtre (figure 4.25-(c)). Pour le cas du disque, on peut observer que le centre de celui-ci est très isolé des autres points sur ce graphe ce qui permet de l’extraire facilement pour construire le squelette.

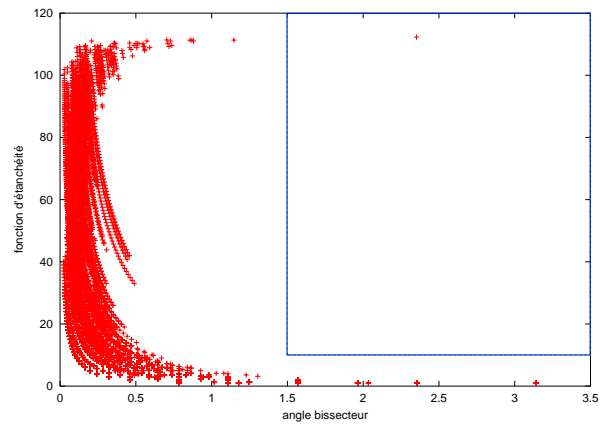
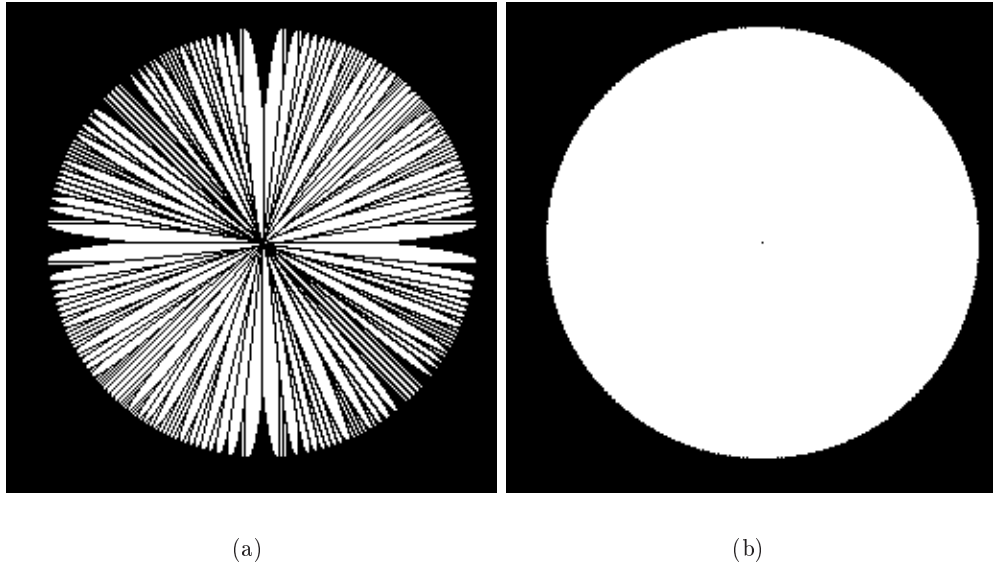
La figure 4.26 illustre le calcul du squelette sur une forme composée de deux disques d’intersection non vide. On peut noter que ces disques n’ont pas exactement le même rayon et les centres ne sont pas exactement sur l’horizontale. Dans la figure 4.26-(c), on notera que le squelette *attendu* se localise assez bien sur le graphe des paramètres. Les formes *hyperboliques* que l’on peut observer dans ces graphes s’expliquent par le fait que si on cherche l’ensemble des points d’un squelette généré par deux points du complémentaire distants de d , nous obtenons la fonctionnelle (ATTALI, 1995) :

$$\rho = \frac{d}{2 \sin \frac{\alpha}{2}}$$

Finalement, les figures 4.27 et 4.28 illustrent le calcul du squelette sur un carré et une forme *humanoïde*.

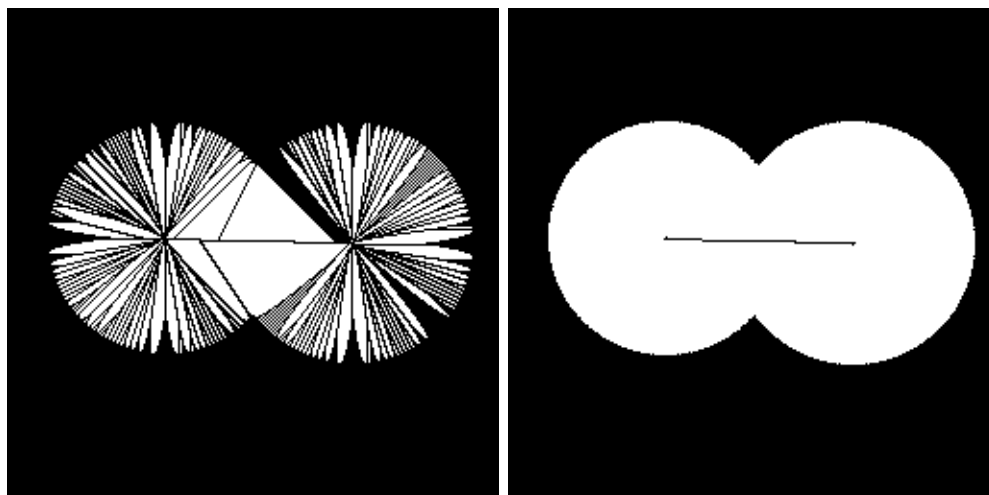
4.3.3 Conclusion et discussion

Nous avons présenté une définition de squelette discret basée sur la notion de diagramme de Voronoï discret. Nous obtenons donc un algorithme d’extraction qui ne nécessite pas de structure de type table de correspondance (*Look-up table*) et qui construit le squelette en temps optimal (linéaire en le nombre de points de l’image).



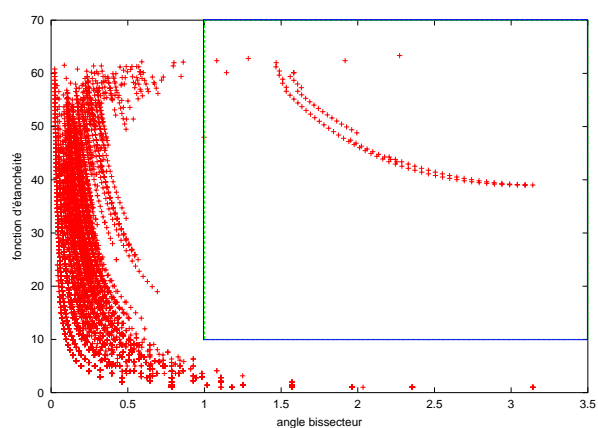
(c)

FIG. 4.25 —: *Squelette d'un disque* : (a) *points du squelette sans simplification*, (b) *squelette après simplification avec $\alpha_0 = 1.5$ et $\rho_0 = 10$* , et (c) *graphe des paramètres $\alpha(s)$ et $\rho(s)$ avec les seuils*.



(a)

(b)



(c)

FIG. 4.26 –: *Squelette de deux disques joints : (a) points du squelette sans simplification, (b) squelette après simplification avec $\alpha_0 = 1$ et $\rho_0 = 10$, et (c) graphe des paramètres $\alpha(s)$ et $\rho(s)$ avec les seuils.*

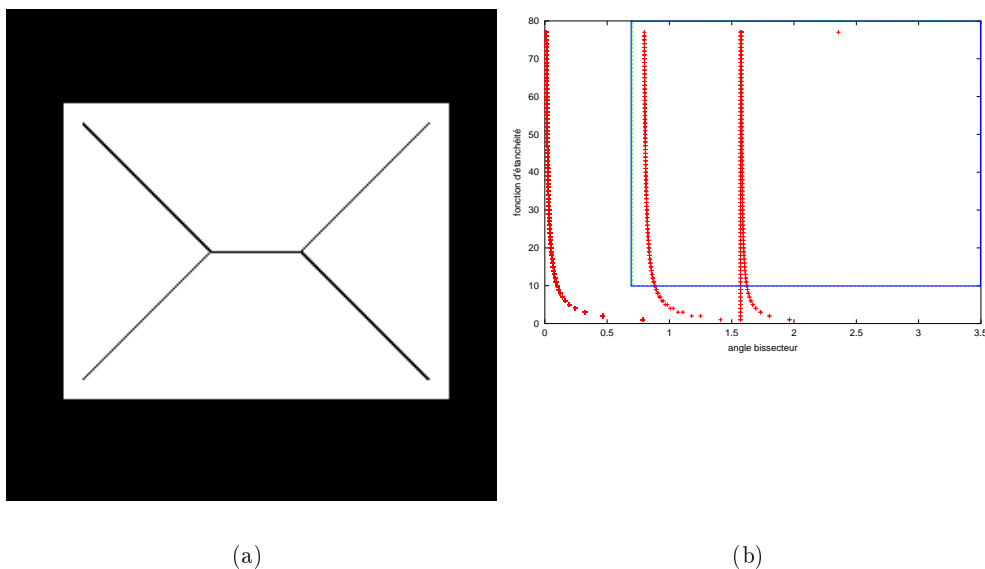


FIG. 4.27 –: *Squelette d'un rectangle : (a) squelette après simplification avec $\alpha_0 = 0.7$ et $\rho_0 = 10$, et (b) graphe des paramètres $\alpha(s)$ et $\rho(s)$ avec les seuils.*

Nous avons aussi utilisé quelques techniques de simplification de squelette sur des critères locaux de ATTALI (1995). Ce travail nous a permis de vérifier la faisabilité pratique de cette technique de construction de squelette.

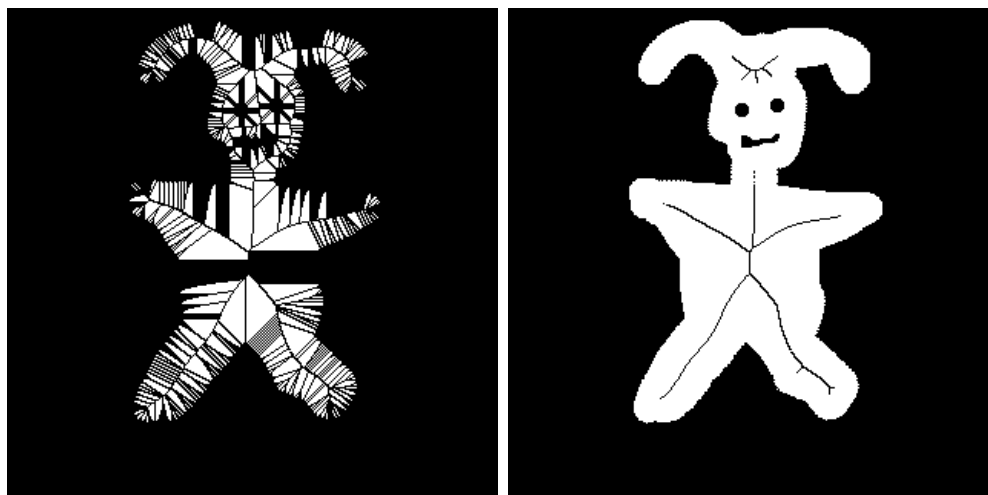
De plus, ce squelette peut s'extraire de la même manière quelle que soit la dimension de l'image et pour des formes de grilles variées. En effet, la construction du diagramme de Voronoï discret, comme nous l'avons vu précédemment, se fait en temps optimal. Les techniques de simplification de squelette présentées s'adaptent aussi en dimension supérieure (voir ATTALI 1995 pour le cas 3D).

Nous avons donc un algorithme optimal en temps pour toute dimension qui permet d'extraire un squelette d'une forme discrète.

Dans le schéma proposé, nous n'avons appliqué que des simplifications locales du squelette sans aucune préservation de quelque propriété topologique. Ce travail de simplification globale apparaît donc comme une perspective importante de cette approche.

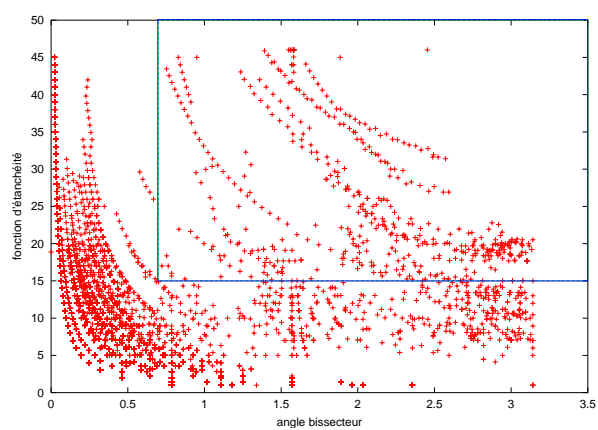
Au niveau des simplifications locales du squelette, de nombreuses autres approches peuvent être envisagées. On peut, par exemple, proposer une simplification par *hysteresis* se référant à des techniques classiques de détection de contour en analyse d'images. Ainsi, un tel processus de simplification manipulerait deux paires de seuils (α_0, ρ_0) et (α_1, ρ_1) et aurait le fonctionnement suivant :

1. si $\alpha(s) > \alpha_1$ et $\rho(s) > \rho_1$, le point du squelette est conservé
2. si $\alpha(s) < \alpha_0$ et $\rho(s) < \rho_0$, le point du squelette est supprimé
3. dans le cas intermédiaire, on regarde s'il existe un chemin d'une certaine taille joignant le point p et un point satisfaisant la première règle. On peut même *diriger* la recherche de ce chemin en regardant dans la direction de la médiatrice du segment $[p_0p_1]$ de la définition 4.8 (voir figure 4.29 pour une illustration dans le graphe des paramètres du nouveau filtre).



(a)

(b)



(c)

FIG. 4.28 – : *Squelette d'une forme synthétique : (a) points du squelette sans simplification, (b) squelette après simplification avec $\alpha_0 = 0.7$ et $\rho_0 = 15$, et (c) graphe des paramètres $\alpha(s)$ et $\rho(s)$ avec les seuils.*

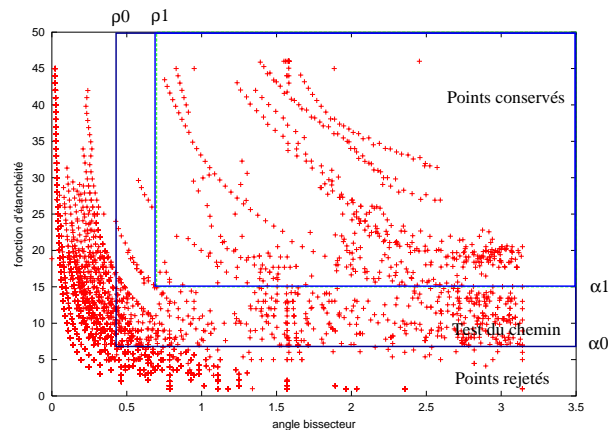


FIG. 4.29 –: Illustration sur le graphe des paramètres de la simplification du squelette par hysteresis.

Sur le plan méthodologique, une perspective intéressante serait la comparaison de ce type de squelette avec les axes médians calculés par table de correspondance. Au delà de cette comparaison, ATTALI ET AL. (1997) ont illustré une extraction de squelette utilisant à la fois le diagramme de Voronoï continu et l'axe médian associés à la forme. Une tel calcul hybride de squelette se basant sur notre diagramme de Voronoï discret nous semble être une perspective importante.

4.4 Visibilité et géodésiques discrètes

Dans cette partie, nous nous intéressons aux notions de visibilité en géométrie discrète et de métrique géodésique. L'intérêt de définir de tels outils est double : d'une part nous augmentons le modèle discret d'un nouvel outil d'analyse, mais nous répondons aussi à un besoin pratique dans de nombreuses applications (par exemple l'angioscopie virtuelle imagerie médicale).

En géométrie algorithmique classique, la notion de visibilité se définit de la manière suivante : étant donné un domaine avec des obstacles polygonaux, un point est *visible* d'un autre si le segment entre ces deux points ne rencontre aucun obstacle. En se basant sur cette définition, nous pouvons définir la notion de chemin géodésique. Considérons deux points et le graphe de visibilité dont les sommets sont les sommets des obstacles polygonaux et les deux points considérés. Les arêtes de ce graphe correspondent à la relation binaire de visibilité, et sont pondérées par la distance euclidienne. Ainsi, le chemin géodésique (ou plus court chemin) entre les deux points correspond au plus court chemin dans le graphe précédent (DE BERG ET AL., 2000; CORMEN ET AL., 1990). La figure 4.30 illustre ces différentes définitions.

En géométrie discrète, nous considérons un domaine dont les obstacles sont des ensembles de pixels connexes. Une première définition de visibilité discrète a été proposée par SOILLE (1991, 1994, 1999). Celle-ci se base sur le tracé de droite de BRESENHAM (1965) : le test de visibilité consiste à tracer le segment de BRESENHAM entre les deux pixels et à vérifier qu'aucun obstacle n'est rencontré. En se basant sur cette définition, SOILLE propose aussi un calcul de chemin géodésique. Nous reviendrons sur cette approche ultérieurement.

Par la suite, LENOIR (1999) propose une approche basée sur des notions de combinatoire et de graphe. Dans le cas 2D, le chemin géodésique entre deux points

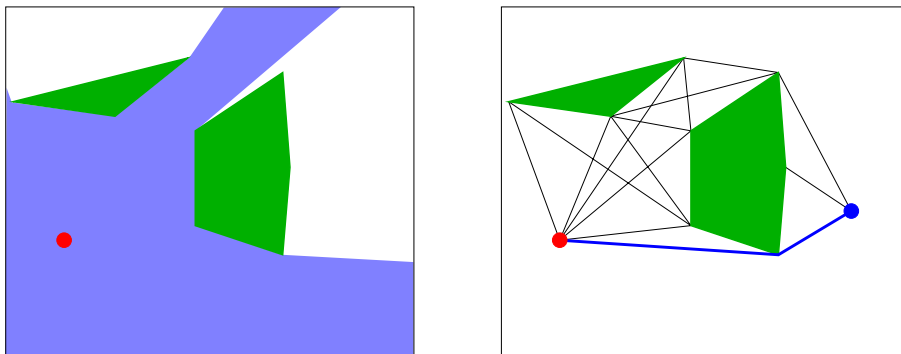


FIG. 4.30 – : *Illustration des définitions : à gauche, tous les points dans la zone bleue sont visibles du point rouge. À droite, construction du graphe de visibilité et du plus court chemin (en gras) entre le point rouge et le point bleu.*

est défini de la manière suivante : on commence par dénombrer tous les chemins 4-connexes joignant les deux points considérés. Ensuite, à chaque pixel visité par un chemin 4-connexe, nous affectons une valeur numérique qui correspond au nombre de chemins qui passent par ce point. Finalement, le plus court chemin correspond au chemin qui maximise la somme de ces valeurs. En d’autres termes, ce chemin géodésique passe par les pixels *les plus empruntés*. Ce processus se généralise très bien sur des surfaces. LENOIR a montré que cette définition de chemin géodésique sur une grille 2D sans obstacle correspond à une droite discrète 2D. Cependant, dans le cas surfacique ou avec des obstacles, aucune équivalence avec des droites ni aucune propriété métrique de la distance ainsi définie n’est proposée. Cette approche à l’avantage de ne se baser que sur le graphe d’adjacence des points et peut donc se généraliser à d’autres grilles régulières.

Dans ce qui suit, nous proposons une définition de visibilité basée sur la notion de droite discrète (voir chapitre 2). Cette proposition permet l’utilisation d’algorithmes efficaces pour le test de visibilité ainsi que pour le calcul de chemins géodésiques.

Dans un premier temps nous présentons les différentes définitions et le calcul de visibilité proposé. Nous décrivons par la suite un algorithme efficace pour le calcul de chemins géodésiques basé sur notre définition de visibilité. Enfin, nous donnons quelques pistes pour la généralisation en dimension 3.

Cette partie correspond aux articles COEURJOLLY (2002) et COEURJOLLY ET AL. (2002b).

4.4.1 Visibilité dans des domaines discrets

4.4.1.1 Notations et définitions

Dans ce qui suit, nous notons \mathcal{D} un domaine discret correspondant à un ensemble de pixels k -connexes. On notera $\bar{\mathcal{D}}$ le complémentaire de \mathcal{D} . Ce complémentaire sera appelé par la suite indifféremment le *fond* ou *l’ensemble des obstacles*. Par la suite, nous considérons un domaine \mathcal{D} 8-connexe.

Par analogie avec le cas polygonal, nous définissons la visibilité discrète de la façon suivante :

Définition 4.9 (Visibilité discrète) *Soient s et t deux pixels de \mathcal{D} , la visibilité discrète est une relation binaire $v \subseteq \mathcal{D} \times \mathcal{D}$ telle que nous avons $v(s, t)$ si et seulement*

s'il existe un segment de droite discrète 8-connexe joignant s et t dont les pixels appartiennent à \mathcal{D} .

Avant de continuer avec le problème d'étiquetage en visibilité, rappelons les liens entre les droites discrètes et l'intersection de bandes dans l'espace des paramètres vus au paragraphe 2.2.2.1.

La discrétisation *Grid Intersect Quantization* (ou GIQ) d'une droite réelle d'équation $y = \alpha x + \beta$ est l'ensemble des points discrets vérifiant :

$$\Delta(\alpha, \beta) = \{(x, y) \in \mathbb{Z}^2 \mid -\frac{1}{2} \leq \alpha x + \beta - y < \frac{1}{2}\}$$

Le choix du processus de discrétisation n'a aucune influence sur la méthode. Nous avons choisi GIQ pour sa propriété *d'approximation au plus proche* de la droite réelle.

Dans l'espace des paramètres, l'équation précédente nous permet de définir l'ensemble des droites réelles dont les discrétisations contiennent un pixel $p(x, y)$:

$$S_p = \{(\alpha, \beta) \in \mathbb{R}^2 \mid -\frac{1}{2} + y \leq \alpha x + \beta < \frac{1}{2} + y\}$$

Ainsi, un pixel de \mathcal{D} définit une *bande* dans l'espace des paramètres délimitée par les contraintes $L_1 : \alpha x + \beta - y \geq -1/2$ et $L_2 : \alpha x + \beta - y < 1/2$. Rappelons que pour savoir si un ensemble de pixels est un morceau de droite discrète, il suffit de vérifier que l'intersection des bandes données par les pixels est non-vide dans l'espace des paramètres. Le domaine résultant de cette intersection définit toutes les droites réelles dont les discrétisations contiennent l'ensemble de pixels considéré (voir l'exemple donné figure 4.31).

On notera par la suite $\mathcal{S}(s, t)$ l'intersection des bandes S_s et S_t , respectivement associées à deux pixels s et t .

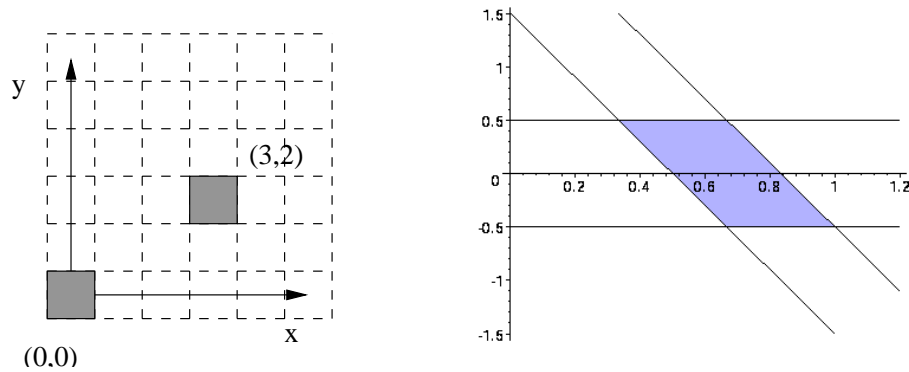


FIG. 4.31 –: Exemple de domaine $\mathcal{S}(s, t)$ des pixels $(0, 0)$ et $(3, 2)$: le domaine $\mathcal{S}(s, t)$ est défini par les inéquations $\{\beta < 1/2, \beta \geq -1/2, \beta < -3\alpha + 5/2, \beta \geq -3\alpha + 3/2\}$.

En ce qui concerne le problème de visibilité entre s et t dans des domaines non-convexes, l'objectif est d'analyser les contraintes liées aux obstacles et aux deux pixels s et t dans l'espace des paramètres.

4.4.1.2 Domaine de visibilité

Soit o un pixel obstacle (*i.e.* $o \in \bar{\mathcal{D}}$). L'ensemble des droites réelles dont la discrétisation ne contient pas le pixel o se décrit simplement dans l'espace des paramètres. En

effet, il nous suffit de considérer les négations des contraintes L_1 et L_2 précédentes. Ainsi, nous associons à un pixel obstacle o les contraintes $\bar{L}_1(o) : \alpha x + \beta - y < -1/2$ et $\bar{L}_2(o) : \alpha x + \beta - y \geq 1/2$.

Maintenant, pour décider si le pixel o empêche la visibilité entre s et t , il nous suffit de tester si le domaine, dans l'espace des paramètres, $L_1(s) \cap L_2(s) \cap L_1(t) \cap L_2(t) \cap (\bar{L}_1(o) \cup \bar{L}_2(o))$ est vide ou non.

Plus généralement, nous considérons un ensemble $\mathcal{O} = \{o_i\}_{i=1..n}$, sous-ensemble de $\bar{\mathcal{D}}$, de pixels obstacles tel que toutes les abscisses de ces points sont comprises entre les abscisses de s et de t (tous les autres obstacles pouvant être supprimés du problème de visibilité entre s et t). Ainsi nous avons :

Lemme 4.3 Soient s et t deux points de \mathcal{D} , s et t sont visibles dans \mathcal{D} si et seulement si :

$$\mathcal{S}(s, t) \cap \left(\bigcap_{i=1..n} (\bar{L}_1(o_i) \cup \bar{L}_2(o_i)) \right) \neq \emptyset$$

La preuve de ce lemme est directement induite de la définition de la visibilité. Le domaine issu de cette intersection est appelé **domaine de visibilité** de s et t (voir figure 4.32).

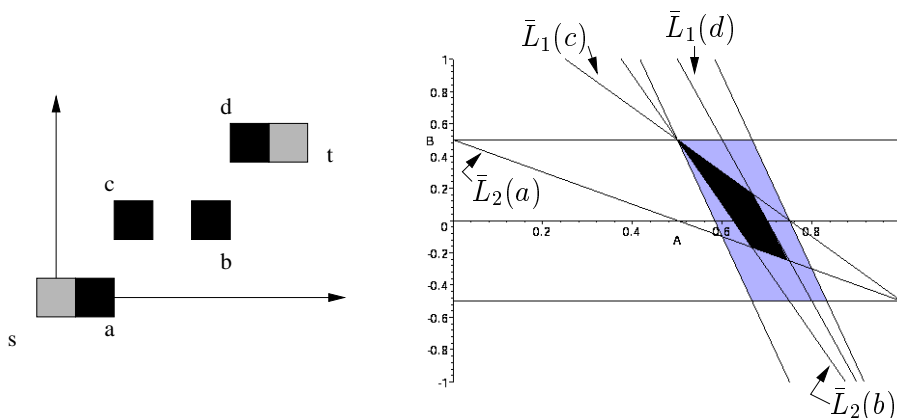


FIG. 4.32 –: *Domaine de visibilité associé à un ensemble de pixels : le domaine en gris clair correspond à $\mathcal{S}(s, t)$ et le polygone noir au domaine de visibilité associé à s, t et aux pixels bloquants a, b, c et d . Les contraintes non tracées correspondent aux contraintes n'intersectant pas $\mathcal{S}(s, t)$.*

Dans ce qui suit, nous cherchons à caractériser l'ensemble des pixels les plus contraignants pour notre problème. Plus formellement, ces pixels sont ceux définis comme suit :

Définition 4.10 (Pixel bloquant) Un pixel o de \mathcal{O} est dit **bloquant** pour le test de visibilité $v(s, t)$ si nous avons :

$$\mathcal{S}(s, t) \cap (\bar{L}_1(o) \cup \bar{L}_2(o)) \neq \mathcal{S}(s, t)$$

et l'abscisse de o est entre les abscisses de s et t .

Nous pouvons présenter un premier lemme :

Lemme 4.4 *Soit o un pixel bloquant pour le test de $v(s, t)$, ou bien le domaine $\mathcal{S}(s, t) \cap (\bar{L}_1(o) \cup \bar{L}_2(o))$ est vide, ou bien il n'a qu'une seule composante connexe.*

Preuve : Nous considérons le domaine $\mathcal{S}(s, t)$ et un pixel bloquant o tel que o, s et t ne sont pas alignés (dans le cas contraire, le domaine de visibilité est vide).

Nous montrons que soit $\bar{L}_1(o)$ soit $\bar{L}_2(o)$ intersecte le domaine $\mathcal{S}(s, t)$ mais jamais les deux. Sans perte de généralité, nous pouvons considérer les différents cas illustrés dans la figure 4.33-(a). Étant donné que l'abscisse de o doit être entre celle de s et t , la pente des contraintes $\bar{L}_1(o)$ et $\bar{L}_2(o)$ doit nécessairement être comprise entre les pentes des contraintes associées à s et t . Nous pouvons donc supprimer les deux premiers cas de la figure 4.33-(a).

De plus, par construction, la hauteur verticale entre L_1 et L_2 est égale à 1. Ainsi, l'intersection en a' entre $\bar{L}_1(o)$ et la droite verticale passant par b implique que l'intersection b' , entre $\bar{L}_2(o)$ et cette même droite verticale, est en dehors du segment $[ab]$ (voir figure 4.33-(b)). De plus, étant donné que la pente de $\bar{L}_2(o)$ est plus grande, en valeur absolue, que celle de la droite (cb) , $\bar{L}_2(o)$ ne peut pas intersecter le domaine $\mathcal{S}(s, t)$.

De la même manière, si $\bar{L}_2(o)$ coupe le domaine, la contrainte $\bar{L}_1(o)$ ne le coupe pas. Le dernier cas de figure de la figure 4.33-(a) n'est donc pas valide.

Finalement, $\mathcal{S}(s, t) \cap (\bar{L}_1(o) \cup \bar{L}_2(o))$ est égal, soit à $\mathcal{S}(s, t) \cap \bar{L}_1(o)$, soit à $\mathcal{S}(s, t) \cap \bar{L}_2(o)$. Ce domaine correspond donc à une intersection de contraintes linéaires, il n'a donc qu'une seule composante connexe qui est de plus convexe.

□

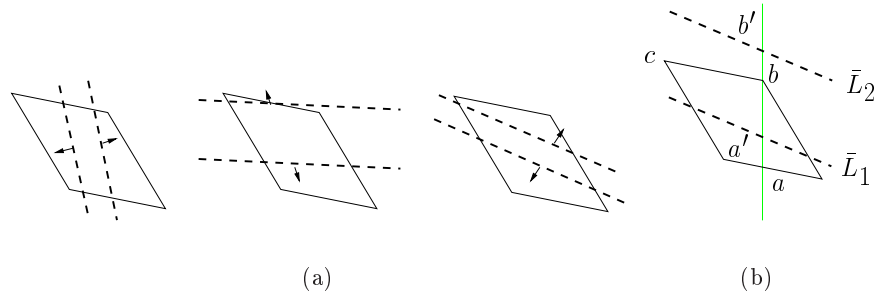


FIG. 4.33 — Illustration de la preuve du lemme 4.4 : (a) différents cas de figure qui introduisent deux composantes connexes et (b) notation pour la preuve du lemme.

D'un point de vue géométrique, si un pixel o est bloquant pour $v(s, t)$ (avec $[st]$ dans le premier octant) et si celui-ci est au dessus du segment $[st]$, seule la contrainte \bar{L}_1 doit être prise en compte. De la même manière, si o est sous $[st]$, seule la contrainte \bar{L}_2 intervient (voir figure 4.32 pour une illustration). Cette remarque nous permet d'avoir une première caractérisation géométrique de cet ensemble de pixels bloquants.

Nous pouvons étendre cette analyse proposée dans COEURJOLLY (2002); COEURJOLLY ET AL. (2002b) par le théorème suivant :

Théorème 4.2 *Étant donné s et t dans \mathcal{D} , le nombre de pixels bloquants suffisant pour évaluer $v(s, t)$ est au plus de deux. Le domaine de visibilité est calculé à partir de $\mathcal{S}(s, t)$ avec au plus quatre contraintes.*

Preuve : Pour plus de clarté, nous choisissons la représentation arithmétique des droites discrètes. Nous avons déjà montré dans le paragraphe 2.2.2.3 les liens entre cette représentation et l'analyse duale.

Sans perte de généralité, soit $[st]$ dans le premier octant et $(a, b) \in \mathbb{Z}^2$ tel que a/b est la pente de $[st]$ (avec $\text{pgcd}(a, b) = 1$). Une droite discrète naïve (8-connexe) qui passe par s et t est composée de l'union de b pointillés consécutifs parmi les $2b - 1$ pointillés :

$$\begin{aligned} ax - by &= -b + 1 \\ &\dots \\ ax - by &= b - 1 \end{aligned}$$

Un pixel obstacle $o(x_o, y_o)$ de reste $r = ax_o - by_o$ invalide tout le pointillé r de l'ensemble $\{-b + 1, \dots, b - 1\}$. De plus, si celui-ci est tel que $r < 0$, il invalide tous les pointillés $\{-b + 1, \dots, r\}$. En effet, nous ne pouvons pas trouver b pointillés consécutifs dans cet ensemble. Ainsi, un autre pixel obstacle dont le reste est dans cet ensemble ne sert plus pour $v(s, t)$ puisqu'il a déjà été exclu par o .

De même, si $r > 0$, tous les pointillés de l'ensemble $\{r, \dots, b - 1\}$ sont invalidés ainsi que tous les pixels obstacles dont le reste est dans cet ensemble.

Ainsi, parmi l'ensemble \mathcal{O} des pixels bloquants, seuls deux pixels nous intéressent :

- parmi les pixels bloquants de restes négatifs, celui dont le reste r_{max} est maximal. Ce point est noté o_{max} .
- Parmi les pixels bloquants de restes positifs, celui dont le reste r_{min} est minimal, noté o_{min} .

Ainsi, il existe une droite discrète si et seulement si il reste b pointillés non invalidés et donc si :

$$r_{min} - r_{max} + 1 \geq b$$

Si nous revenons au domaine de visibilité $\mathcal{S}(s, t)$, quatre points sont nécessaires. Ils correspondent aux pixels bloquants de reste r_{max} et r_{min} d'abscisse minimale et maximale entre s et t . De plus, grâce au lemme précédent, une seule contrainte, \bar{L}_1 ou \bar{L}_2 , est à prendre en compte. Ainsi, le domaine de visibilité est exactement calculé à partir de $\mathcal{S}(s, t)$ et des quatre contraintes issues des pixels bloquants précédent.

□

Une illustration de ce théorème est donnée dans la figure 4.32 : les pixels c et d sont tous les deux sur le pointillé d'abscisse r_{max} pour le segment $[s, t]$. De même, les points a et b sont sur le pointillé r_{min} .

4.4.1.3 Algorithme de calcul de visibilité

Dans ce paragraphe, nous présentons un algorithme efficace permettant de calculer la classe d'équivalence de la relation de visibilité associée à un pixel source s . En d'autres termes, étant donné un point s de \mathcal{D} , nous voulons marquer tous les pixels de \mathcal{D} qui sont visibles de s . Pour cela, nous proposons deux algorithmes : le premier

correspond au calcul de la classe de visibilité basée sur les définitions ci-dessus. Le second algorithme se base sur une version modifiée de la définition de visibilité mais permet un étiquetage plus efficace du domaine.

L'algorithme 22 présente donc une implémentation directe des définitions et théorèmes précédents. Cet algorithme se base sur un parcours en largeur du domaine \mathcal{D} , implémenté par la structure de file FIFO. À chaque étape de l'algorithme, nous maintenons une liste des obstacles \mathcal{O} et nous étiquetons en visibilité tous les pixels de \mathcal{D} rencontrés.

Algorithme 22 Classe d'équivalence de visibilité à partir d'une source s

```

CALCUL_VISIBILITÉ( $s, \mathcal{D}$ )
1: Soit  $Q$  une file
2: Soit  $\mathcal{O}$  une liste de pixels obstacles
3:  $Q.AJOUTER(s)$ 
4: tant que  $Q$  non vide faire
5:    $t = Q.PREMIER\_ELEMENT()$ 
6:   pour tout  $p$ , 8-voisin de  $t$  non étiqueté fermé ou visible faire
7:     si  $p$  est un obstacle alors
8:        $O.AJOUTER(p)$ 
9:     sinon
10:      Soit  $(a, b, c, d)$  les quatre contraintes de  $\mathcal{O}$  décrit dans le théorème 4.2
11:      Évaluer  $v(s, p)$  avec  $(a, b, c, d)$ 
12:      si  $p$  est visible de  $s$  alors
13:        Marquer  $p$  comme étant visible
14:         $Q.AJOUTER(p)$ 
15:      sinon
16:        Marquer  $p$  comme étant fermé
17:      fin du si
18:    fin du si
19:  fin du pour
20: fin du tant que

```

Analysons la complexité de cet algorithme, pour cela nous notons n le nombre de pixels de \mathcal{D} . Dans l'algorithme 22, le maintien de la file FIFO et de la liste des obstacles se fait en temps constant à chaque étape. Grâce au théorème 4.2, le test de visibilité $v(s, p)$ se fait en $O(1)$ si nous connaissons le quadruplet (a, b, c, d) . Finalement, le point clef de la complexité est donné par la ligne 10 de cet algorithme qui consiste à extraire de \mathcal{O} le quadruplet de pixels bloquants. Or, aucune structure de données ne nous permet d'accéder à ces éléments rapidement dans le cas de la visibilité présenté ci-dessus. Ainsi, si nous notons m le nombre d'obstacles, la complexité globale de cet algorithme est en $O(nm)$, ce qui est assez coûteux, surtout dans un objectif de calcul de géodésiques.

Nous proposons maintenant un second algorithme basé sur une définition légèrement différente de la visibilité. Celle-ci nous permet de mettre en place une structure de données efficace. Nous définissons ainsi la notion de visibilité restreinte :

Définition 4.11 (Visibilité restreinte) Soient s et t deux pixels de \mathcal{D} , la **visibilité restreinte** est une relation binaire $v^* \subseteq \mathcal{D} \times \mathcal{D}$ telle que nous avons $v^*(s, t)$ si et seulement si il existe une demi-droite réelle qui passe par s et dont la discrétisation contient t et aucun pixel de $\bar{\mathcal{D}}$.

Dans cette définition, nous contraignons donc l'ensemble des droites à passer

exactement par s . Cette version de la visibilité est un affaiblissement de la précédente puisque la relation binaire que l'on obtient n'est plus symétrique. Cependant, cette définition nous permet de proposer une structure de données et un algorithme permettant un étiquetage en visibilité très efficace.

Supposons donc que tous les pixels obstacles sont triés dans un ordre polaire de centre s dans le sens trigonométrique. Nous avons donc :

Proposition 4.3 *Soient un ensemble de pixels obstacles triés en ordre polaire de centre s et un point t . Nous notons u le successeur de t dans le tri polaire et l son prédécesseur. Nous avons alors :*

$$v^*(s, t) \Leftrightarrow \mathcal{S}^*(s, t) \cap \bar{L}_1(u) \cap \bar{L}_2(l) \neq \emptyset$$

où \mathcal{S}^* correspond au domaine des droites vérifiant la contrainte de la visibilité restreinte. Ce domaine correspond à un segment dans l'espace des paramètres.

La preuve de cette proposition est directement induite de la définition de v^* ainsi que des lemmes précédents. Remarquons enfin que le tri polaire des pixels obstacles peut se calculer entièrement en nombres entiers (voir par exemple CORMEN ET AL. 1990).

Nous pouvons maintenant présenter l'algorithme du calcul de visibilité (voir algorithme 23).

Algorithme 23 Classe d'équivalence de visibilité restreinte à partir d'une source s

CALCUL_VISIBILITÉ_RESTREINTE(s, \mathcal{D})

```

1: Soit  $Q$  une file FIFO
2: Soit  $\mathcal{O}$  une liste de pixels obstacles triés en polaire de centre  $s$ 
3: AJOUT_FILE( $s, Q$ )
4: tant que  $Q$  non vide faire
5:    $t = \text{PREMIER\_ELEMENT}(Q)$ 
6:   pour tout  $p$ , 8-voisin de  $t$  non étiqueté fermé ou visible faire
7:     si  $p$  est un obstacle alors
8:       AJOUT_TRI_POLAIRE( $p, \mathcal{O}$ )
9:     sinon
10:      Soit  $u$  le successeur de  $p$  dans le tri polaire  $\mathcal{O}$ 
11:      Soit  $l$  le prédécesseur de  $p$  dans le tri polaire  $\mathcal{O}$ 
12:      Évaluer  $v^*(s, p)$  avec  $\mathcal{S}^*(s, p) \cap \bar{L}_1(u) \cap \bar{L}_2(l) \neq \emptyset$ 
13:      si  $p$  est visible de  $s$  alors
14:        Marquer  $p$  comme étant visible
15:        AJOUT_FILE( $p, Q$ )
16:      sinon
17:        Marquer  $p$  comme étant fermé
18:      fin du si
19:    fin du pour
20:  fin du tant que
21: fin du tant que

```

Dans cet algorithme, le test de visibilité s'effectue aussi en temps constant mais cette fois, l'extraction, à partir de \mathcal{O} , des pixels bloquants *intéressants* pour notre définition de visibilité se fait en $O(\log(m))$ au lieu de $O(m)$. De plus, le maintien de la liste triée, c'est-à-dire le coût de l'insertion d'un obstacle, se fait lui aussi en $O(\log(m))$. Finalement, le coût global de cet algorithme est en $O(n \log(m))$.

Nous pouvons encore optimiser le coût moyen de ce dernier avec la remarque suivante : lors du parcours en largeur du graphe, quand nous insérons les 8-voisins de t dans la structure, nous mémorisons le cône (s, u, l) qui correspond à la localisation de t dans \mathcal{O} . Ainsi, lorsque nous dépilerons les fils de t , nous mettrons à jour leur localisation à partir de (s, u, l) . De cette façon, nous obtenons un coût de mise-à-jour très faible.

Dans une analyse au pire cas, cette optimisation ne permet pas de réduire la complexité, cependant, si nous considérons des ensembles pixels obstacles assez compacts dans \mathcal{D} , le coût effectif de cette mise à jour sera très faible.

4.4.2 Plus court chemin discret et métrique géodésique discrète

Nous pouvons maintenant définir les notions de chemins géodésiques discrets basées sur le calcul de visibilité.

4.4.2.1 Définitions et état de l'art

Dans un premier temps, rappelons les deux grandes approches pour les définitions de métriques discrètes vues précédemment :

- approches par masque (métriques de chanfrein) : on considère un ensemble de déplacements élémentaires pondérés ;
- approches basées sur une gestion de déplacement vectoriels (d_x, d_y) .

Dans le cas de la métrique géodésique, les approches basées sur des masques offrent une solution très efficace. En effet, à partir de \mathcal{D} et des déplacements élémentaires considérés, nous pouvons construire un graphe d'adjacence pondéré sur lequel nous pouvons appliquer un algorithme classique de plus court chemin dans un graphe comme celui de DIJKSTRA (PIPER et GRANUM, 1987). Pour implémenter cet algorithme de plus court chemin, nous pouvons utiliser des **files de priorité** (ou *bucket lists* en anglais). Cette structure est classique en algorithmique (voir par exemple CORMEN ET AL. 1990) et son utilisation pour le calcul de distance géodésique par masque a été proposée par VERWER ET AL. (1989). Celle-ci nous permet d'avoir un algorithme à coût uniforme pour l'étiquetage en distance géodésique.

Cette structure, illustrée figure 4.34, est construite de la façon suivante : nous considérons une collection de files FIFO indicées par un entier d . Pour notre problème, tous les pixels se trouvant dans la file d'indice d sont à une distance d de notre point d'origine. Le principe de l'algorithme de plus court chemin basé sur cette technique est le suivant : l'algorithme commence par ajouter le point s dans la file d'indice 0. Ensuite, à chaque étape on dépile un pixel p , encore non visité, de la file d'indice minimum non vide, notée d_{min} , et pour chaque voisin n de p , nous ajoutons n à la file d'indice $d_{min} + weight(p, n)$. $Weight(p, n)$ correspond au poids associé au déplacement $p\vec{n}$ dans le masque de chanfrein utilisé.

Cette structure permet donc une implémentation très efficace de l'algorithme de DIJKSTRA. De plus, VERWER ET AL. (1989) proposent une gestion des allocations des files très efficaces, par exemple, nous pouvons réutiliser les files de manière circulaire une fois que celles-ci sont vides.

Dans une approche basée sur la distance euclidienne, MOREAU (1995) propose un étiquetage du domaine en maintenant des listes de pixels du front de propagation. Ce processus de propagation se base sur une construction incrémentale de cercles à

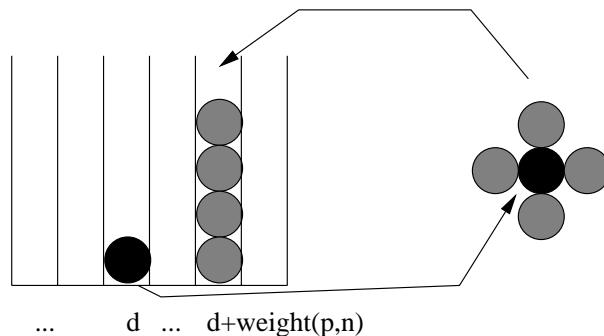


FIG. 4.34 –: Structure de files de priorité : ensemble de files FIFO indicées par la distance au pixel source s .

rayon croissant. Ainsi, quand aucun obstacle n’est rencontré, le processus construit des cercles de rayon croissant en étiquetant le pixels par la distance euclidienne au centre du cercle. Si un obstacle est rencontré on subdivise le cercle en arcs avec la possibilité d’initialiser de nouveaux centres d’arcs de cercle. La structure de données de cet algorithme n’est cependant pas évidente à manipuler et des processus coûteux en temps peuvent apparaître lors de la fusion ou de la subdivision d’arcs.

En se basant sur l’utilisation de sa transformée en distance euclidienne par propagation (voir paragraphe 4.2.2.1), CUISENAIRE (1999) propose un algorithme de calcul de distance géodésique que l’on peut qualifier de local. En effet, étant donné un certain rayon r , tous les pixels dans la boule euclidienne de rayon r centré sur le pixel source s sont étiquetés par cette distance euclidienne. Ensuite, le processus de propagation est itéré avec les pixels non étiquetés sur le front de propagation. Ainsi, si le rayon choisi est grand, la distance calculée dans les domaines où il y a peu d’obstacles sera une bonne approximation de la distance euclidienne. Cependant, l’approximation sera mauvaise dans les parties fortement contraintes. Ainsi, la qualité de l’approximation est fonction du domaine \mathcal{D} et du rayon r choisis.

Enfin, SOILLE (1991, 1994, 1999) propose l’utilisation de la notion de visibilité pour l’estimation de chemins géodésiques. Dans le paragraphe suivant, nous présentons une approche similaire mais utilisant le calcul efficace de visibilité présenté ci-dessus.

4.4.2.2 Algorithme d’étiquetage en distance géodésique

L’idée principale de l’algorithme est la suivante : pour tous les points visibles depuis le point source s , l’étiquetage en distance géodésique est trivial puisqu’il correspond exactement à la distance euclidienne calculée à partir du vecteur de déplacement (d_x, d_y) . Par contre, si un point p n’est pas visible depuis la source, nous initialisons un nouveau processus d’étiquetage en visibilité à partir de p . Ainsi, nous étiquetons en distance les pixels visités, visibles de p , par la distance entre p et la source originale s à laquelle nous ajoutons la distance euclidienne entre ces points et p .

Définissons tout d’abord la notion de chemin géodésique discret :

Définition 4.12 (Chemin géodésique discret) *Un chemin géodésique discret entre deux points s et p de \mathcal{D} est une séquence de pixels $\{p_i\}_{i=0..n+1} \in \mathcal{D}$, avec $p_0 = s$ et $p_{n+1} = t$, telle que :*

$$v(p_i, p_{i+k}) \quad \text{pour } k = \{-1, 0, 1\} \quad \text{et } i = 1..n$$

et telle que la distance géodésique $d_{geodes}(s, t)$ est minimale. Cette distance est définie par :

$$d_{geodes}(s, t) = \sum_{i=0}^n d_{euc}(p_i, p_{i+1})$$

où $d_{euc}(a, b)$ représente la distance euclidienne classique.

En d'autres termes, un chemin géodésique discret est une séquence de segments de droite discrète de longueur minimale joignant s et t . La métrique géodésique proposée pour cette séquence de segments correspond exactement à celle utilisée pour estimer la longueur d'une courbe discrète basée sur une segmentation en droites discrètes (voir paragraphe 5.4).

Afin d'obtenir un algorithme d'étiquetage efficace, nous utilisons la structure de files de priorité de VERWER ET AL. dont les indices sont donnés par $\lceil d_{geodes}(s, p) \rceil$. Cette métrique est consistante avec l'algorithme de VERWER car elle satisfait l'inégalité triangulaire (MOREAU, 1995) :

$$\text{for } a, b, c \in \mathbb{R} \quad a + b \geq c \Rightarrow \lceil a \rceil + \lceil b \rceil \geq \lceil c \rceil$$

Toujours dans un souci d'efficacité, nous utilisons la visibilité restreinte $v^*(s, t)$. Pour chaque pixel p dans la structure de file, nous mémorisons : ses coordonnées, les coordonnées du point source s_i duquel p est visible et la distance géodésique $d_{geodes}(s, s_i)$. Nous maintenons aussi une structure triée de pixels obstacles associés à chaque propagation en visibilité.

L'algorithme 24 présente la méthode proposée.

4.4.3 Analyse expérimentale et discussion

Dans notre analyse expérimentale, nous avons appliqué l'algorithme précédent sur un certain nombre de domaines binaires. Ainsi, la figure 4.35 présente les résultats de l'étiquetage en distance géodésique avec les métriques d_4 , d_8 et la métrique géodésique proposée. Les distances sont représentées avec une palette de niveaux de gris réduite afin de mieux observer les fronts de propagations.

Dans la figure 4.36, au lieu d'étiqueter par la distance géodésique, tous les pixels de même couleur sont dans la même classe d'équivalence de visibilité. Ces illustrations nous permettent de proposer une solution au problème *jouet* classique de placement du nombre minimal de gardes dans un musée. En effet, chaque couleur correspond à la zone de visibilité de chaque garde et le domaine \mathcal{D} correspond au musée. Ainsi, si le premier garde est fixé (source s du problème) nous sommes capable de calculer le nombre minimal de gardes pour couvrir tout le domaine et de placer ceux-ci.

La figure 4.37 présente une application de ces algorithmes en imagerie médicale. L'objectif est ici de mesurer la longueur de vaisseaux sanguins dans des images angiographique segmentées.

Enfin, si nous appliquons cet algorithme d'étiquetage en distance géodésique à partir de plusieurs sources, le résultat obtenu est très proche d'une transformée en distance euclidienne de ces points sources. Cependant, étant donné que notre processus de propagation du domaine se fait localement, nous ne pouvons pas éviter les erreurs de type DANIELSON présentés dans les paragraphes précédents.

Algorithme 24 Étiquetage en distance géodésique à partir d'une source s .

CALCUL_GÉODÉSIQUE(s, \mathcal{D})

- 1: Soit $Bucket[i]$ un ensemble de files FIFO
- 2: Soit $\mathcal{O}[i]$ un ensemble de listes de pixels bloquants triés en polaire
- 3: Soit d_{min} l'indice minimal des files non vides initialisé à 0
- 4: AJOUT_FILE($s, Bucket[d_{min}]$)
- 5: **tant que** il reste une file non vide **faire**
- 6: **si** la file $Bucket[d_{min}]$ est vide **alors**
- 7: ajuster d_{min}
- 8: **fin du si**
- 9: $t = \text{PREMIER_ELEMENT}(Bucket[d_{min}])$
- 10: **pour tout** p , 8-voisin de t non étiqueté **fermé** ou **visible** **faire**
- 11: Soit i l'indice de la liste d'obstacles associé à la source s_i de t
- 12: **si** p est un obstacle **alors**
- 13: AJOUT_TRI_POLAIRE($p, \mathcal{O}[i]$)
- 14: **sinon**
- 15: Soient u et l la localisation de p dans le tri polaire $\mathcal{O}[i]$
- 16: Évaluer $v^*(s, p)$ avec $\mathcal{S}^*(s, p) \cap \bar{L}_1(u) \cap \bar{L}_2(l) \neq \emptyset$
- 17: **si** p est visible de s **alors**
- 18: Marquer p comme étant **visible**
- 19: Calculer la distance géodésique d' de n
- 20: **si** $d' > d_{min}$ **alors**
- 21: AJOUT_FILE($p, Bucket[[d'x]]$)
- 22: **fin du si**
- 23: **sinon**
- 24: Marquer p comme étant **fermé**
- 25: **fin du si**
- 26: **fin du pour**
- 27: **fin du tant que**
- 28: **fin du tant que**

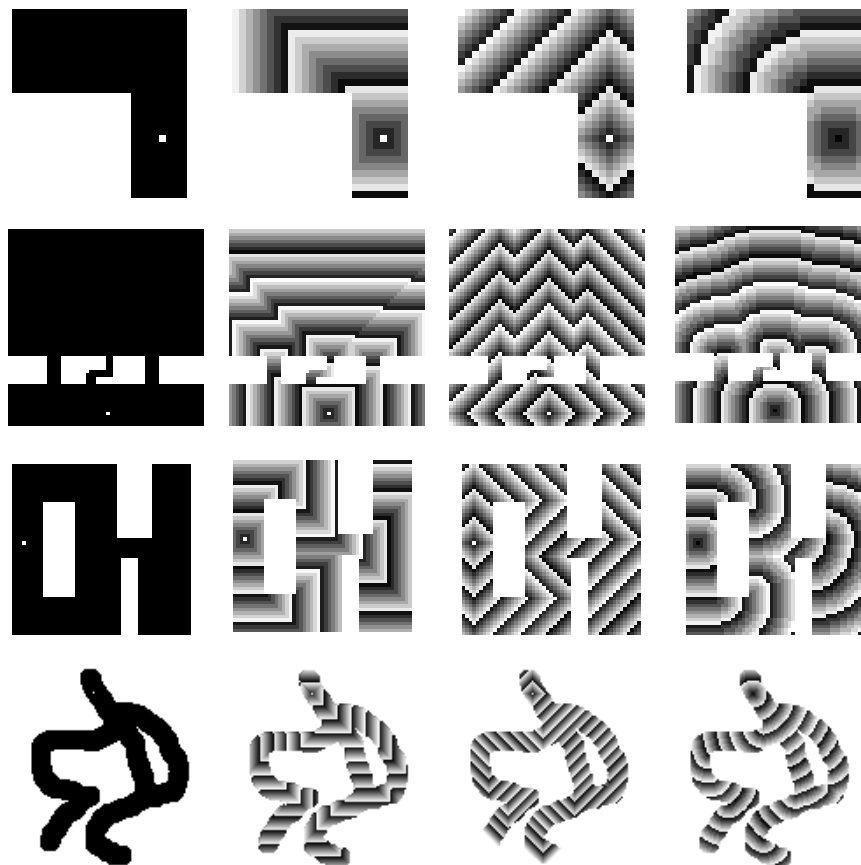


FIG. 4.35 – : *Analyse expérimentale de l'étiquetage en distance géodésique : de gauche à droite, les domaines discrets et le point source (pixel blanc isolé), l'étiquetage avec la distance d_8 , avec la distance d_4 et enfin avec la distance géodésique proposée.*

4.4.4 Vers une extension en 3D

Une extension naturelle de ce travail consisterait en une analyse de la visibilité et des chemins géodésiques en dimensions supérieures. Nous nous intéressons ici au cas de la dimension 3 et considérons un domaine \mathcal{D} de voxels 26-connexes.

Dans le cas des approches basées sur des distances de chanfrein, des algorithmes efficaces peuvent être proposés pour l'étiquetage en distance géodésique. En effet, nous nous ramenons de toute façon à un problème de graphe (KIRYATI et SZÉKELY, 1993; SANNITI DI BAJA et SVENSSON, 2000).

Dans notre approche de chemins géodésiques basés sur la visibilité, nous pouvons reprendre les mêmes définitions : on dira que deux voxels sont visibles s'il existe un segment de droite discrète 26-connexe tel que tous les voxels sont dans \mathcal{D} , passant par les deux voxels. Nous renvoyons le lecteur au chapitre 2.3 pour plus de détails sur les droites discrètes 3D. Cependant, l'étiquetage en visibilité dans des domaines non-convexes 3D est plus complexe que dans le cas 2D. Pour illustrer cela, regardons une version 3D de la visibilité restreinte. Nous cherchons donc à décrire les droites réelles passant exactement par le centre d'un voxel s et dont la discrétisation contient un voxel t de \mathcal{D} et aucun point de $\bar{\mathcal{D}}$.

Sans perte de généralité, nous nous plaçons dans le premier 48^{ème} d'espace, c'est-à-dire nous considérons $\vec{st} = (a, b, c)^T$ tel que $a \geq b \geq c > 0$. Ainsi, une droite qui

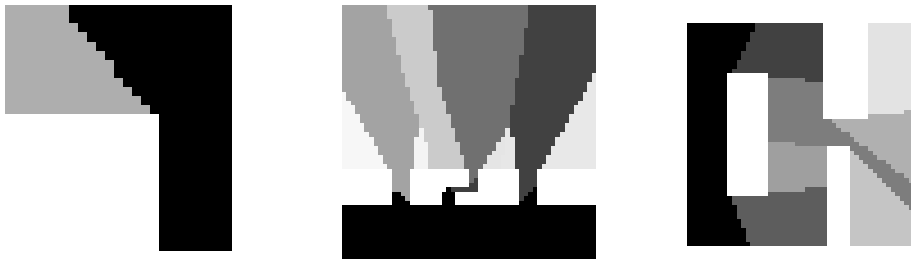


FIG. 4.36 –: *Diagramme de visibilité discrète : tous les pixels de même couleur sont dans la même classe d'équivalence de visibilité, les domaines et points sources sont ceux de la figure 4.35.*

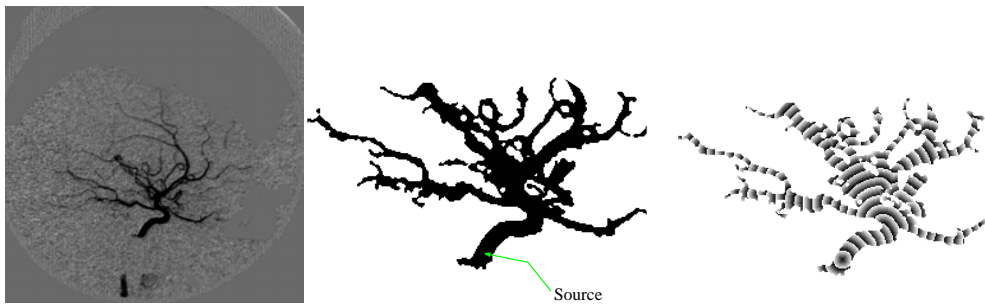


FIG. 4.37 –: *Application de la distance géodésique en imagerie médicale : de gauche à droite, l'image angiographique de vaisseaux sanguins, le domaine binaire issu d'une segmentation et l'étiquetage en distance géodésique.*

passer par s et qui intersecte le carré de longueur de côté 1 centré en t dans le plan (Oyz) est solution au problème de visibilité restreinte (voir figure 4.38-(a)). Si maintenant nous ajoutons des obstacles dans le même $48^{\text{ème}}$ d'espace, nous construisons des pyramides similaires et le point t sera visible de s s'il existe une droite passant par s , qui intersecte le carré centré en t mais ne coupe pas les carrés associés aux obstacles.

Maintenant, projetons l'ensemble de ces carrés sur le plan parallèle à (Oyz) contenant t en utilisant la projection perspective de centre s . Ainsi, t sera visible de s si et seulement si le carré associé à t n'est pas entièrement recouvert par ceux associés aux voxels obstacles (voir figure 4.38-(b)).

D'un point de vue algorithmique, même le cas de la visibilité restreinte n'est pas évident. En effet, l'écriture d'un algorithme efficace qui teste la couverture d'un carré par d'autres n'est pas trivial. De plus aucune structure géométrique évidente des voxels obstacles ne peut être mise en place en 3D pour cette visibilité.

Nous pouvons cependant proposer une première solution en généralisant en 3D l'approche de SOILLE (1991, 1994, 1999). Dans ce cas, le test de visibilité est basé sur un tracé de droite discrète 26-connexe en utilisant un des algorithmes présenté dans le paragraphe 2.3). Ainsi, en utilisant ce test de visibilité et les mêmes structures que pour le cas 2D, nous obtenons un algorithme d'étiquetage en distance géodésique

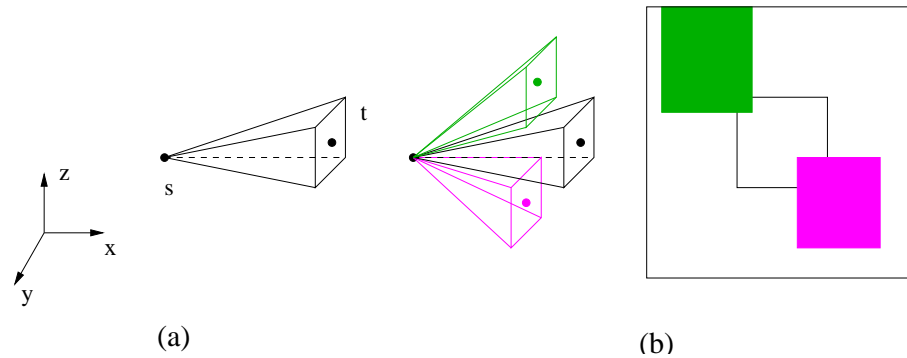


FIG. 4.38 — *Illustration de la visibilité restreinte en 3D : (a) domaines de visibilité restreinte dans l'espace primal et (b) illustration du test de visibilité par projection dans le plan (Oyz) contenant t des carrés associés aux voxels bloquants.*

dans des domaines 3D. Cette méthode n'est cependant pas très efficace puisque la complexité est en $O(nd)$ où n est le nombre de voxels de \mathcal{D} et d son diamètre (longueur du plus long segment de droite contenu dans \mathcal{D}).

Les figures 4.39 et 4.40 présentent un étiquetage en distance géodésique de domaines 3D non convexes en utilisant une palette de niveaux de gris réduite.

Si nous considérons maintenant une surface discrète comme un ensemble de voxels connexes, l'algorithme précédent nous permet de définir la notion de géodésiques surfaciques. En effet, les voxels d'une telle surface discrète nous donnent le domaine \mathcal{D} . Les voxels obstacles correspondent au complémentaire de \mathcal{D} . La figure 4.41 illustre ces calculs de géodésiques surfaciques.

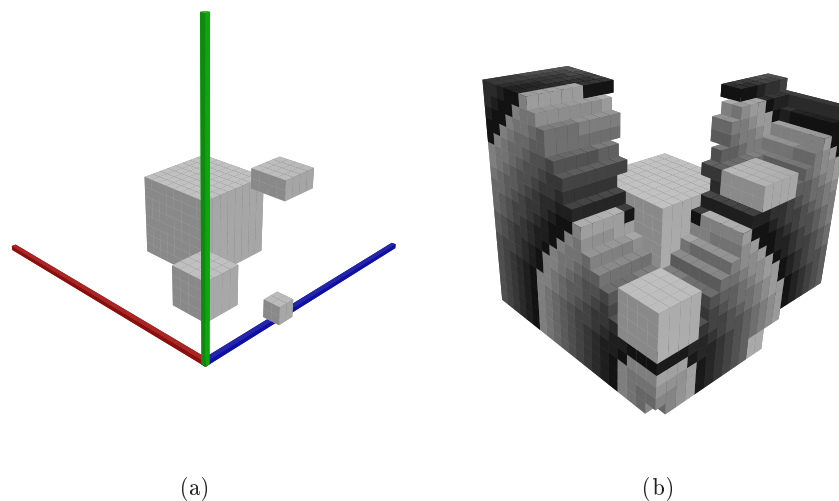


FIG. 4.39 — *Exemple d'étiquetage en distance géodésique 3D : (a) domaine 3D défini par les obstacles gris et (b), étiquetage en distance géodésique de la première classe de visibilité uniquement. Le voxel source est à l'origine du repère.*

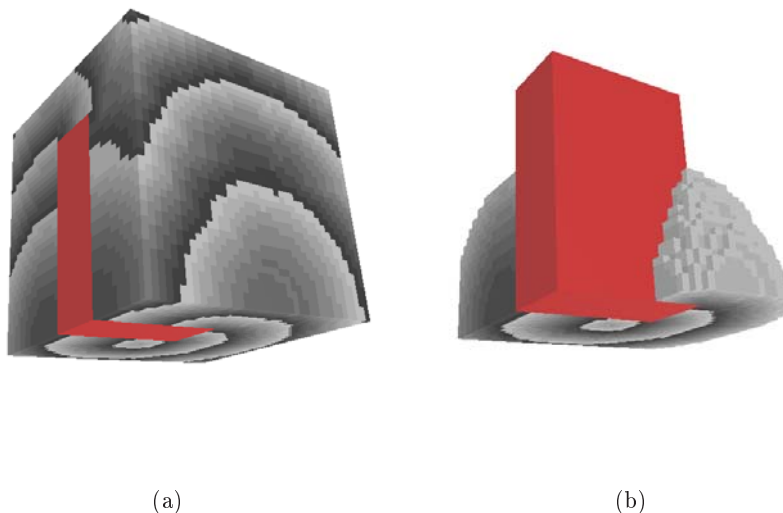


FIG. 4.40 — Exemple d'étiquetage en distance géodésique 3D : (a) étiquetage en distance géodésique complet et (b) seuillage de la distance pour mieux observer les structures internes du front d'onde.

4.4.5 Conclusion et perspectives

Nous avons présenté un test de visibilité discrète basé sur un objet discret classique (la droite discrète). Lors de notre étude de la visibilité discrète, nous avons aussi analysé le comportement de ces droites discrètes dans le cas de domaines non-convexes.

Par la suite, nous avons présenté plusieurs algorithmes permettant : de décider si deux pixels sont visibles étant donné un ensemble de pixels obstacles, de décrire l'ensemble des pixels d'un domaine visibles à partir d'un point. Sur le plan algorithmique nous avons présenté un algorithme d'étiquetage en visibilité efficace dont la complexité est en $O(n \log(m))$ (où n est le nombre de pixels dans \mathcal{D} et m le nombre de pixels obstacles). Nous avons ensuite présenté une définition de chemin géodésique discret basé sur la notion de visibilité ainsi qu'un algorithme de calcul.

Nous nous sommes aussi intéressés au cas tridimensionnel en présentant un premier algorithme ainsi que quelques pistes vers un algorithme efficace.

4.5 Conclusion sur ce chapitre

Dans ce chapitre, nous nous sommes intéressés aux métriques associées à la grille discrète et à l'utilisation de celles-ci pour l'analyse de formes. L'intérêt des descriptions obtenues à l'aide de la transformée en distance euclidienne et du squelette de forme a été montré depuis un certain temps. A la suite de la rédaction de ce chapitre, nous nous sommes aperçus qu'un article récent (MAURER JR. ET AL., 2001) présentait un résultat analogue pour le calcul de la transformée à partir du diagramme de Voronoï discret. L'algorithme présenté a été développé indépendamment des travaux de MAURER JR. ET AL.. Cependant, nous donnons dans ce chapitre des démonstrations et illustrations plus exhaustives. Par contre, la construction d'un squelette basé sur le diagramme de Voronoï discret est originale à notre connaissance.

En ce qui concerne la propagation géodésique, elle peut être utilisée soit di-

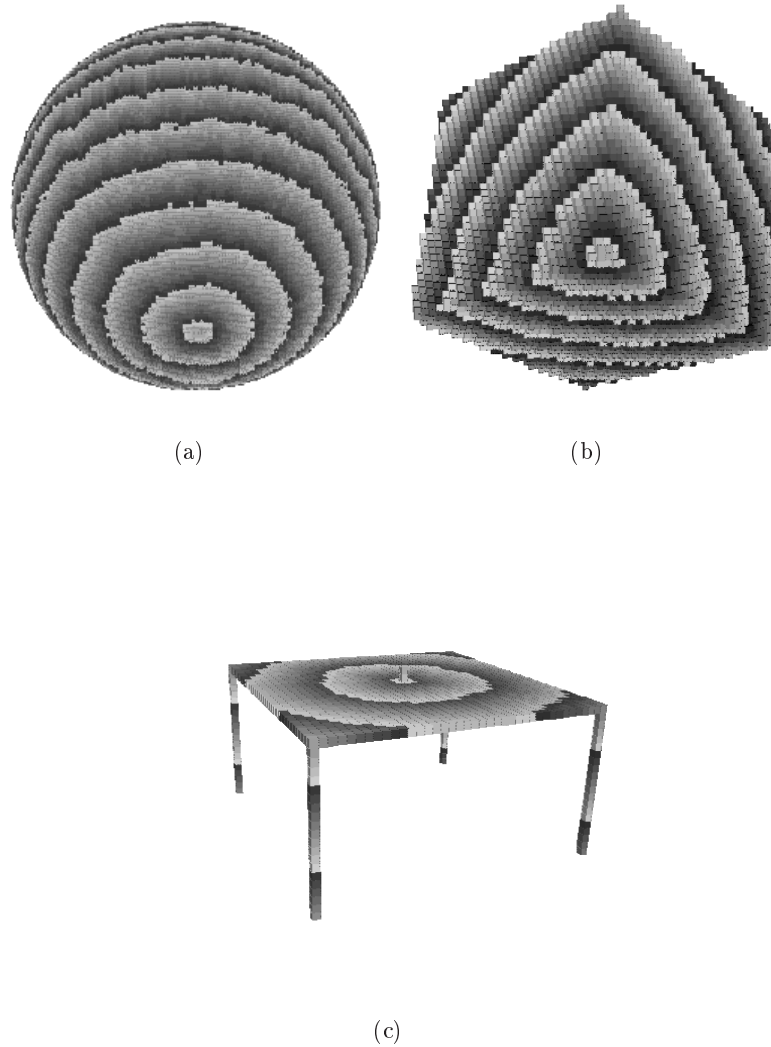


FIG. 4.41 —: *Géodésiques surfaciques* : (a) *étiquetage en distance géodésique de la surface d'une sphère*, (b) *de la surface d'un cube* et (c) *d'une table (le point source est situé au milieu du plateau)*.

rectement comme source d'information métrique mais permet aussi une description globale de la forme en considérant les notions de transformée en distance géodésique (SOILLE, 1991, 1999). De plus, la possibilité de construire un *voisinage géodésique* autour d'un voxel d'une surface discrète permet l'écriture d'algorithmes d'estimation de normales et de courbure Gaussienne (voir paragraphes 5.3.2 et 5.5.3.2).

Les perspectives de ces travaux sont nombreuses. Pour le squelette de formes basé sur le diagramme de Voronoï discret, l'étude présentée nous a permis de vérifier la pertinence d'une telle approche. Il nous faut cependant situer un peu plus précisément celle-ci dans la bibliographie et surtout présenter des calculs d'extraction en dimensions quelconques.

Pour la propagation géodésique, nous souhaitons comparer cette approche discrète aux approches basées sur une analyse différentielle (SETHIAN, 1996b,a). Enfin, nous souhaitons aussi poursuivre l'analyse en dimension 3 afin de réduire la complexité de l'algorithme proposé.

Chapitre 5

Mesures sur des objets discrets

Sommaire

5.1	Introduction	163
5.2	Contexte théorique : la convergence asymptotique	163
5.3	Tangentes et normales	165
5.3.1	Tangentes et normales discrètes d'un contour 2D	165
5.3.2	Normales d'une surface discrète	167
5.3.2.1	Approches volumiques	168
5.3.2.2	Approches surfaciques	174
5.3.2.3	Estimateur optimal en temps avec convergence asymptotique	177
5.4	Longueur et aire	178
5.4.1	Longueur d'une courbe discrète 2D	178
5.4.1.1	Approches basées sur des métriques locales	179
5.4.1.2	Approches basées sur des approximations polygonales	180
5.4.1.3	Approches basées sur l'intégration des normales discrètes	183
5.4.1.4	Évaluation comparative	184
5.4.1.5	Remarques sur la plate-forme de tests	187
5.4.2	Longueur d'une courbe discrète 3D	190
5.4.3	Aire d'une surface discrète	194
5.4.3.1	Solutions existantes	194
5.4.3.2	Estimation d'aire de surface discrète par intégration de normales	195
5.4.3.3	Analyse expérimentale	196
5.4.3.4	Conclusion sur l'estimation d'aire d'une surface discrète	199
5.5	Courbures d'objets discrets	199
5.5.1	Quelques définitions mathématiques	199
5.5.2	Courbure de courbes discrètes	203
5.5.2.1	État de l'art	203
5.5.2.2	Estimation du cercle osculateur discret	204
5.5.2.3	Estimation de courbure avec convergence asymptotique	208
5.5.2.4	Courbures de courbes discrètes 3D	210
5.5.3	Courbures de surfaces discrètes	212

5.5.3.1	Courbure Moyenne	212
5.5.3.2	Courbure Gaussienne	215
5.6	Conclusion	220

5.1 Introduction

Dans ce chapitre, nous nous plaçons dans le cadre de l'analyse d'objets basée sur le calcul de mesures euclidiennes sur les contours. En effet, une approche classique de reconnaissance de formes se base sur l'association, au contour de l'objet, d'un vecteur contenant des informations numériques (longueur, aire, graphe de courbure...). On parle alors de la *signature* du contour.

Dans ce qui suit, nous nous intéressons à la création d'un tel vecteur d'attributs dans le cas des courbes et des surfaces discrètes. Nous présentons donc des algorithmes permettant d'estimer la longueur d'une courbe discrète ou l'aire d'une surface discrète, la normale en un point d'une courbe ou d'une surface et enfin les différentes courbures associées à ces objets.

Ces différents estimateurs seront évalués, à la fois numériquement dans des analyses expérimentales, mais aussi sur le plan théorique avec la notion de convergence asymptotique. Pour une meilleure lisibilité de ce chapitre, nous avons placé les preuves de convergence asymptotique dans l'annexe B.

Après avoir présenté ce cadre théorique, nous décrivons tout d'abord les différents estimateurs pour le calcul des normales. Remarquons que la logique voudrait que l'on commence par les estimateurs de longueur ou d'aire, ensuite ceux de normales et enfin les calculs de courbure. Cependant, nous verrons en quoi l'estimation des normales est primordiale pour le calcul de longueur d'une courbe discrète ou d'aire d'une surface.

Afin de faciliter la lecture de ce chapitre, les différentes preuves de convergence asymptotique sont données dans l'annexe B.

5.2 Contexte théorique : la convergence asymptotique

Lorsqu'un estimateur d'une mesure euclidienne est proposé sur une donnée discrète, la justification de celui-ci peut se faire de deux manières différentes : la première justification consiste à montrer la cohérence de l'estimateur par rapport aux propriétés déjà existantes dans le modèle. Une autre justification consiste à prouver la convergence asymptotique de l'estimateur. Dans ce cas, l'estimateur est comparé à la mesure euclidienne approximée par le biais d'un processus de discrétisation sur une grille dont la résolution tend vers l'infini. Un estimateur aura alors la propriété de *convergence asymptotique* si l'erreur commise entre la mesure estimée et la mesure euclidienne tend vers 0 quand la résolution de la grille augmente.

Plus formellement et dans le cas bidimensionnel, nous considérons une courbe euclidienne $\gamma : [0, 1] \rightarrow \mathbb{R}^2$ et une grille orthogonale dans le plan euclidien \mathbb{R}^2 dont la distance entre les points est donnée par $0 < \theta \leq 1$. Nous appelons r la résolution de la grille donnée par $r = 1/\theta$, on parle alors de r -grille. Étant donné un processus de discrétisation de la courbe γ sur la grille, noté $D_r(\gamma)$, nous définissons ainsi un objet multi-grille paramétré par r . Soit \mathcal{F} un descripteur sur la courbe euclidienne γ (la longueur par exemple) et un estimateur \mathcal{E} de la même mesure mais sur le discrétisé $D_r(\gamma)$, nous avons :

Définition 5.1 (Convergence asymptotique) *L'estimateur \mathcal{E} converge asymptotiquement vers \mathcal{F} si et seulement si :*

$$|\mathcal{E}(D_r(\gamma)) - \mathcal{F}(\gamma)| \leq \epsilon(r) \quad \text{avec} \quad \lim_{r \rightarrow \infty} \epsilon(r) = 0$$

L'ordre $O(1/\epsilon(r))$ correspond à la vitesse de la convergence.

L'intérêt d'une telle propriété est double :

- sur le plan théorique, cette propriété nous permet une comparaison formelle entre une mesure continue et une estimation discrète ;
- d'un point de vue pratique, cette convergence nous permet de garantir de bons résultats numériques quelque soit la taille de l'objet discret analysé, ce qui est très important dans une approche multi-échelle par exemple.

Une illustration historique de cette validation remonte aux grecs de l'antiquité : ARCHIMÈDE et LIU HUI (voir par exemple VAN DER WAERDEN 1983) cherchaient à estimer le périmètre d'un cercle, l'objectif principal étant en fait l'estimation du nombre π . Ils utilisaient une approximation basée sur des polygones réguliers à n côtés avec $n_m = 3, 6, 12, 24, 48, 96, \dots$ (voir figure 5.1). Plus précisément, on suppose que ρ est le rayon du cercle et $n_m = 3 \times 2^m$ avec $m \geq 0$. Nous considérons le polygone régulier interne P_{n_m} , possédant n_m arêtes. La longueur p_m d'une arête est donnée par la formule de récurrence :

$$p_{m+1} = \rho \sqrt{2\rho^2 - \rho \sqrt{4\rho^2 - p_m^2}}$$

Ainsi, le périmètre de P_{n_m} est égal à $n_m \cdot p_m$ et donc :

$$\kappa(n_m) = |n_m \cdot p_m - 2\pi\rho| \approx \frac{2\pi\rho}{n_m}, \quad \text{pour } m \geq 1.$$

La fonction $\kappa(n_m)$ définit la vitesse de convergence de l'estimation du périmètre du cercle de rayon ρ en fonction du nombre de côtés du polygone régulier. Dans ce cas, l'estimation du périmètre d'un cercle par des polygones réguliers converge asymptotiquement avec une vitesse en $O(\frac{1}{n_m})$ (voir figure 5.1).

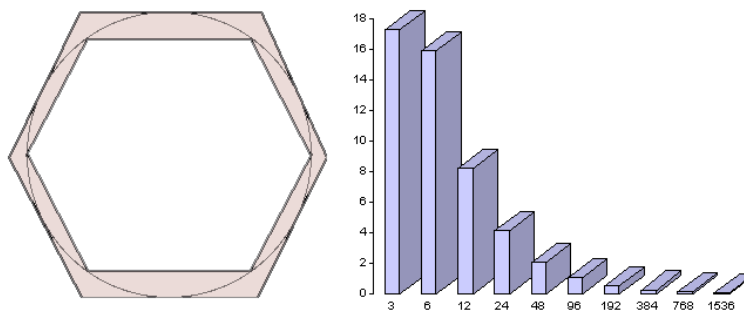


FIG. 5.1 –: (gauche) Polygones réguliers interne et externe d'un cercle pour $n_m = 6$, (droite) pourcentage d'erreur de l'estimation du périmètre avec le polygone régulier interne pour n croissant.

Pour revenir au problème d'estimation du nombre π , cette technique a permis d'obtenir les bornes suivantes pour $n_m = 96$:

$$223/71 < \pi < 220/70$$

En plus de la preuve formelle de la convergence d'un estimateur, nous souhaitons avoir une analyse expérimentale de comportement d'algorithme implémentant l'estimateur. Pour cela, nous devons considérer une famille d'objets multi-grilles la plus hétérogène possible. Bien évidemment, il faut que nous puissions calculer la mesure euclidienne $\mathcal{F}(\gamma)$ sur chacun des objets proposés.

Par exemple, la figure 5.2 présente les différents objets multi-grilles pris en compte pour l'estimation de longueur de courbes discrètes (voir paragraphe 5.4.1). Ces objets ont été choisis car ils illustrent beaucoup de cas concrets : courbes polygonales convexes, courbes euclidiennes non-convexes etc... La discrétisation de ces formes sur une grille de taille variable ainsi que la prise en compte de rotations de ces objets sur la grille nous permettent d'avoir un jeu de tests assez complet.



FIG. 5.2 — *Objets multi-grilles proposés dans KLETTE ET AL. (1999) pour l'analyse d'estimateurs de longueurs.*

Dans le cas tridimensionnel, d'autres objets sont considérés comme nous le verrons dans le paragraphe 5.4.3.

Dans ce qui suit, nous proposons plusieurs estimateurs en 2D et 3D. Nous essayons, autant que possible, de vérifier formellement la propriété de convergence asymptotique. Dans le cas de mesures ponctuelles comme les normales ou la courbure, on dira qu'un estimateur est convergent asymptotique si il y a convergence de l'estimation en tout point.

5.3 Tangentes et normales

5.3.1 Tangentes et normales discrètes d'un contour 2D

Nous considérons dans ce paragraphe le problème de l'estimation de la tangente et de la normale en un point d'une courbe discrète 2D.

Dans une utilisation directe de ces informations, VIALARD (1996b,a) propose l'utilisation de la tangente en tout point pour le calcul du chemin euclidien associé à une courbe discrète : ce chemin correspond à un lissage réversible du contour discret. Plus précisément, chaque sommet initial du contour discret va engendrer un sommet du chemin euclidien, contenu dans un carré de taille 1 centré sur le sommet discret. Ainsi, VIALARD propose l'utilisation d'une notion de tangente discrète pour le positionnement des sommets du chemin euclidien.

Ce calcul de tangente permet aussi la construction d'autres estimateurs de mesures euclidiennes sur des courbes discrètes. De manière intuitive, si nous dérivons la fonction de variation d'angle des tangentes, nous obtenons un estimateur de courbure et si nous intégrons cette information, nous pouvons définir un estimateur de longueur de la courbe discrète considérée. Nous reviendrons plus tard sur ces estimateurs.

D'un point de vue bibliographique, la définition de tangente discrète a été proposée initialement par TROESCH (1993) mais a ensuite été formalisée et analysée par VIALARD (1996b,a); BRAQUELAIRE et VIALARD (1999) :

Définition 5.2 (Tangente discrète) *La tangente discrète en un point p d'une*

courbe discrète est le plus long segment de droite discrète qu'il est possible de reconnaître, centré en p .

Remarquons que la définition précédente est valide quel que soit le type de courbe discrète considérée (ensemble de pixels ou approche cellulaire), il suffit d'adapter la définition de droite discrète au type de courbe manipulé.

Nous pouvons définir très simplement la notion de normale discrète en un point d'une courbe :

Définition 5.3 (Normale discrète) *La normale discrète en un point p d'une courbe discrète est le vecteur orthogonal à la tangente discrète en p dont le sens est donné par une orientation de la courbe discrète.*

D'un point de vue algorithmique, le calcul de la tangente en un point se fait grâce, par exemple, à l'algorithme 5 de reconnaissance de droite discrète modifié par VIALARD. L'idée est ici de tester l'appartenance à une droite discrète en ajoutant symétriquement des pixels à gauche et à droite (voir illustration figure 5.3). C'est pour effectuer cet ajout symétrique que VIALARD (1996a) a modifié l'algorithme initial de DEBLED-RENNESON (1995).

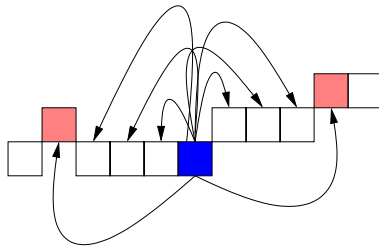


FIG. 5.3 — Calcul de la tangente discrète par ajout symétrique de pixels et test d'appartenance à une droite discrète. Les pixels colorés terminent la tangente.

Si nous implémentons directement cet algorithme pour le calcul des tangentes discrètes en tout point de la courbe discrète, la complexité obtenue est en $O(n^2)$ où n est le nombre de points discrets de la courbe.

Cependant, FESCHET et TOUGNE (1999) ont proposé un algorithme permettant de réduire le coût en un temps linéaire, ce qui est optimal pour le problème. L'idée principale est d'utiliser les paramètres de la tangente discrète en un point de la courbe pour calculer les paramètres de la tangente des points adjacents.

La figure 5.4 illustre le cœur de cet algorithme : supposons connus les paramètres de la tangente en p et soit q le premier point à droite rejeté lors de la reconnaissance symétrique de segment discret. Calculons ensuite le plus long segment de droite discrète à partir de q vers p , nous notons l le premier point rejeté de ce processus et m le milieu du segment $[lq]$ ¹. Ainsi, FESCHET et TOUGNE (1999) prouvent que, pour tous les pixels compris entre p et m (m exclu), les paramètres des tangentes seront exactement les mêmes que ceux de p .

Ainsi, basé sur cette propriété, ils proposent un algorithme incrémental permettant de mettre à jour les positions des points q , l et m lorsque l'on passe du point p au point à sa droite. Les auteurs montrent que cette mise à jour se fait en temps

¹. si le segment $[lq]$ contient un nombre pair de pixels, m est choisi tel qu'il soit plus proche de q que de l

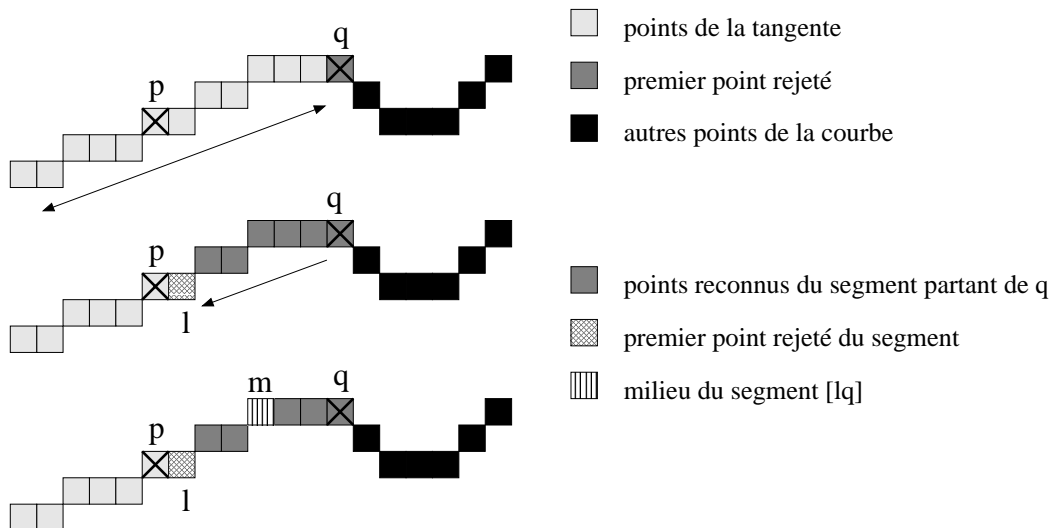


FIG. 5.4 – : Illustration de l'algorithme de FESCHET et TOUGNE (1999) pour le calcul en temps optimal des tangentes discrètes.

constant ce qui permet de calculer toutes les tangentes discrètes en tout point de la courbe en $O(n)$ où n est le nombre de points de cette courbe discrète. Cet algorithme permet une construction des normales en tout point avec la même complexité.

En ce qui concerne la convergence asymptotique, nous avons le théorème suivant :

Théorème 5.1 *Soit Γ la famille des courbes C^2 dans \mathbb{R}^2 à courbure bornée (plus quelques hypothèses techniques). L'estimateur de normales discrètes basé sur la notion de tangente discrète de VIALARD converge asymptotiquement.*

La preuve de ce théorème est donnée dans l'annexe B (théorème B.3).

Pour conclure ce paragraphe, nous avons présenté un calcul de normales d'une courbe discrète qui a la propriété de convergence asymptotique et dont l'algorithme d'estimation est optimal en temps. Nous verrons par la suite que ce résultat est primordial pour de nombreux autres estimateurs.

5.3.2 Normales d'une surface discrète

L'estimation des vecteurs normaux associés à une surface est un problème très important en géométrie discrète. Une des applications privilégiées est la visualisation de volumes binaires. L'idée est de représenter la surface d'un objet discret sous forme lisse en utilisant des techniques d'ombrage se basant sur les normales (ombrage de GOURAUD ou de PHONG par exemple).

Par la suite, nous verrons comment utiliser ces normales surfaciques pour le calcul de nombreux autres estimateurs comme par exemple le calcul de l'aire d'une surface discrète ou le calcul de courbures surfaciques (Gaussienne et Moyenne).

Dans ce qui suit, nous allons présenter les différentes approches de calcul de normales existantes dans la littérature, ainsi que notre contribution à ce problème.

Pour présenter l'état de l'art du calcul de normales, nous reformulons la classification proposée par YAGEL ET AL. (1992). Ainsi, nous pouvons distinguer deux grandes classes d'algorithmes :

Approches volumiques : ces méthodes utilisent des informations volumiques

comme la transformée en distance ou encore la configuration des voxels de l'objet dans un certain voisinage.

Approches surfaciques : dans ce cas, seuls des traitements sur les voxels de la surface sont considérés. Sur le plan algorithmique, ces approches ont l'avantage de manipuler une surface discrète plutôt qu'un volume.

Dans cette bibliographie, nous ne détaillerons pas les approches utilisant des informations de niveaux de gris d'une image non segmentée pour estimer la normale (voir MONGA ET AL. 1994 par exemple).

Quelle que soit l'approche choisie, l'estimation des normales peut se décomposer en deux grandes étapes (YAGEL ET AL., 1992) :

1. Construction des *contextes* : en chaque voxel de la surface discrète, on appelle *contexte* le voisinage autour du voxel qui ne contient pas de discontinuités C^1 .
2. Estimation des normales à partir des contextes : une fois les contextes calculés en chaque voxel, une approximation de la normale est construite à partir des différentes informations contenues dans ce contexte.

Le problème réside principalement dans le calcul de ces contextes. Pour certains algorithmes que nous présentons dans la suite et que nous appelons *locaux*, le contexte en un point est un voisinage de taille fixe. D'autres algorithmes, dits *globaux*, proposent un calcul adaptatif de ce contexte. Enfin, des approches hybrides existent, par exemple un processus adaptatif peut être utilisé pour estimer la normale à partir du contexte de taille fixe.

5.3.2.1 Approches volumiques

Dans ces approches, le volume discret est utilisé pour localiser les différentes discontinuités afin de construire les contextes.

Une première approche que nous présentons a été proposée par THÜRMER et WÜTHRICH (1998). Cet algorithme considère un contexte donné par le parallélépipède centré autour du voxel considéré de taille n_s et un masque de poids w de taille n_s . L'estimation de la normale $(n_x, n_y, n_z)^T$ est donnée par la formule :

$$\begin{aligned} n_x &= \sum_{i,j,k \in \{-n_s \dots n_s\}} \frac{w(i,j,k)\sigma(i,j,k)i}{\sqrt{i^2 + j^2 + k^2}} \\ n_y &= \sum_{i,j,k \in \{-n_s \dots n_s\}} \frac{w(i,j,k)\sigma(i,j,k)j}{\sqrt{i^2 + j^2 + k^2}} \\ n_z &= \sum_{i,j,k \in \{-n_s \dots n_s\}} \frac{w(i,j,k)\sigma(i,j,k)k}{\sqrt{i^2 + j^2 + k^2}} \end{aligned}$$

avec $\sigma(i,j,k) = 1$ si le voxel (i,j,k) est dans l'objet et 0 sinon. Le vecteur $(n_x, n_y, n_z)^T$ est par la suite normalisé.

Cette approche n'utilise pas entièrement le volume de l'objet mais uniquement une épaisseur de celui-ci donnée par n_s . Pour une estimation aussi précise que possible des normales, il faut trouver à la fois la taille de voisinage et le masque des

pois qui minimisent l'erreur d'approximation sur un ensemble d'objets discrets représentatifs de l'application. Par exemple, THÜRMER et WÜTHRICH (1998) propose une analyse expérimentale basée sur une distribution de plans discrets pour estimer les paramètres de la formule précédente.

Nous présentons une solution un peu particulière qui devrait être classée parmi les approches surfaciques mais qui constitue une base à de nombreuses approches surfaciques. Ainsi, GORDON et REYNOLDS (1985) se placent dans le cas de la visualisation de volume discret et cette approximation utilise la carte des profondeurs (**Z-buffer**) pour estimer la normale des voxels visibles depuis un certain point de vue. Plus précisément, si nous notons $z = f(x, y)$ la surface issue du **Z-buffer**, la normale est donnée par le gradient :

$$\nabla z = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, -1 \right)$$

Ainsi, une fois le **Z-buffer** calculé à partir des voxels à la surface de l'objet, on utilise des calculs de dérivées partielles pour extraire les normales associées aux voxels visibles. Cette approche a cependant un problème majeur lorsqu'il y a des parties concaves dans l'objet qui ne se retrouvent pas dans le **Z-buffer**. Elles peuvent, en effet, engendrer des erreurs de calcul.

Cependant, cette approche peut aussi être vue comme une version simplifiée des calculs basés sur une analyse de la carte de distance. L'idée est la suivante : nous représentons la surface discrète comme une iso-potentielle $f(x, y, z) = 0$ d'une certaine fonction f . Ensuite, si nous sommes capables de construire une telle fonction, la normale en tout point de la surface sera donnée par le gradient de f :

$$\nabla f = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z} \right)$$

Une manière simple de construire cette fonction est de considérer la transformée en distance de la surface discrète (voir illustration en 2D figure 5.5).

Ainsi, une estimation de la normale basée sur ce principe se décompose en trois grandes étapes :

1. calcul de la transformée en distance,
2. application de filtres dérivatifs pour l'estimation des dérivées partielles,
3. construction de la normale sur cette carte de gradients.

Cet algorithme a été implémenté par FLIN ET AL. (2001) avec un processus adaptatif pour construire les normales à partir du gradient de la transformée en distance (noté GTD par la suite). L'idée est de détecter les discontinuités de la surface dans cette carte de distance. Pour cela, si nous considérons un certain voisinage N dans la carte GTD interne à l'objet, la normale à la surface de l'objet est calculée par moyennage des vecteurs dans le voisinage N de la GTD en maintenant trois critères :

- un critère d'angle afin de détecter des discontinuités sur la surface. Ce critère se base sur un écart angulaire, entre la normale estimée à un certain instant et le vecteur de GTD analysé, qui doit être inférieur à 30° (cette valeur correspond aux structures cristallines importantes dans les échantillons de neige, voir paragraphe A.3).

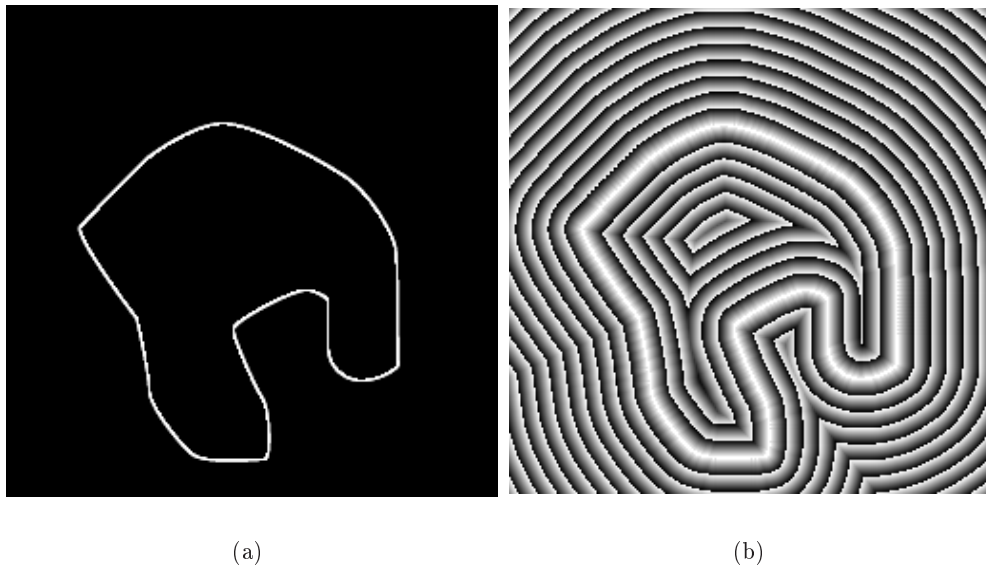


FIG. 5.5 – : *Analyse de la carte de distance pour l'estimation des normales : (a) bord d'un objet discret 2D et (b) utilisation de la transformée en distance euclidienne (algorithme 20).*

- Un critère de symétrie, l'ensemble des vecteurs de la GTD utilisés devant être symétriques autour de la normale estimée.
- Un critère de connexité des points considérés dans le GTD.

Dans COEURJOLLY ET AL. (2002a), nous avons évalué cet estimateur de normale. Une première validation a consisté en un rendu d'objets discrets dont la fonction d'illumination est dirigée par les normales (ombrage de GOURAUD). Ainsi, la figure 5.6 permet de comparer visuellement les rendus entre notre algorithme et l'approche classique par Marching-Cubes (voir paragraphe 2.4.2.4). La figure 5.7 présente quelques résultats de rendu d'un échantillon de neige (voir paragraphe A.3). Nous pouvons remarquer une bonne préservation des différentes discontinuités des objets.

Cet estimateur n'a pas la propriété de convergence asymptotique puisqu'il nécessite une taille de voisinage comme paramètre. Cependant, nous avons évalué expérimentalement la mesure d'erreur. Ainsi, dans la figure 5.8, nous avons estimé les normales sur un ensemble de sphères discrètes de rayons croissants. Pour chaque sphère, nous traçons l'erreur moyenne dans l'estimation des normales en degré ainsi que l'écart type de cette estimation. Ces résultats montrent un très bon comportement de cet estimateur local mais adaptatif basé sur le GTD.

Enfin, nous pouvons proposer une nouvelle approche dans laquelle nous utilisons directement le diagramme de Voronoï discret présenté dans l'algorithme 20. En effet, si nous associons, à tout point de l'objet, le vecteur unitaire vers le site (point du fond) le plus proche, nous obtenons directement une très bonne estimation du GTD sans aucune application de filtres dérivatifs toujours difficiles à paramétrer. La figure 5.9 illustre ce calcul.

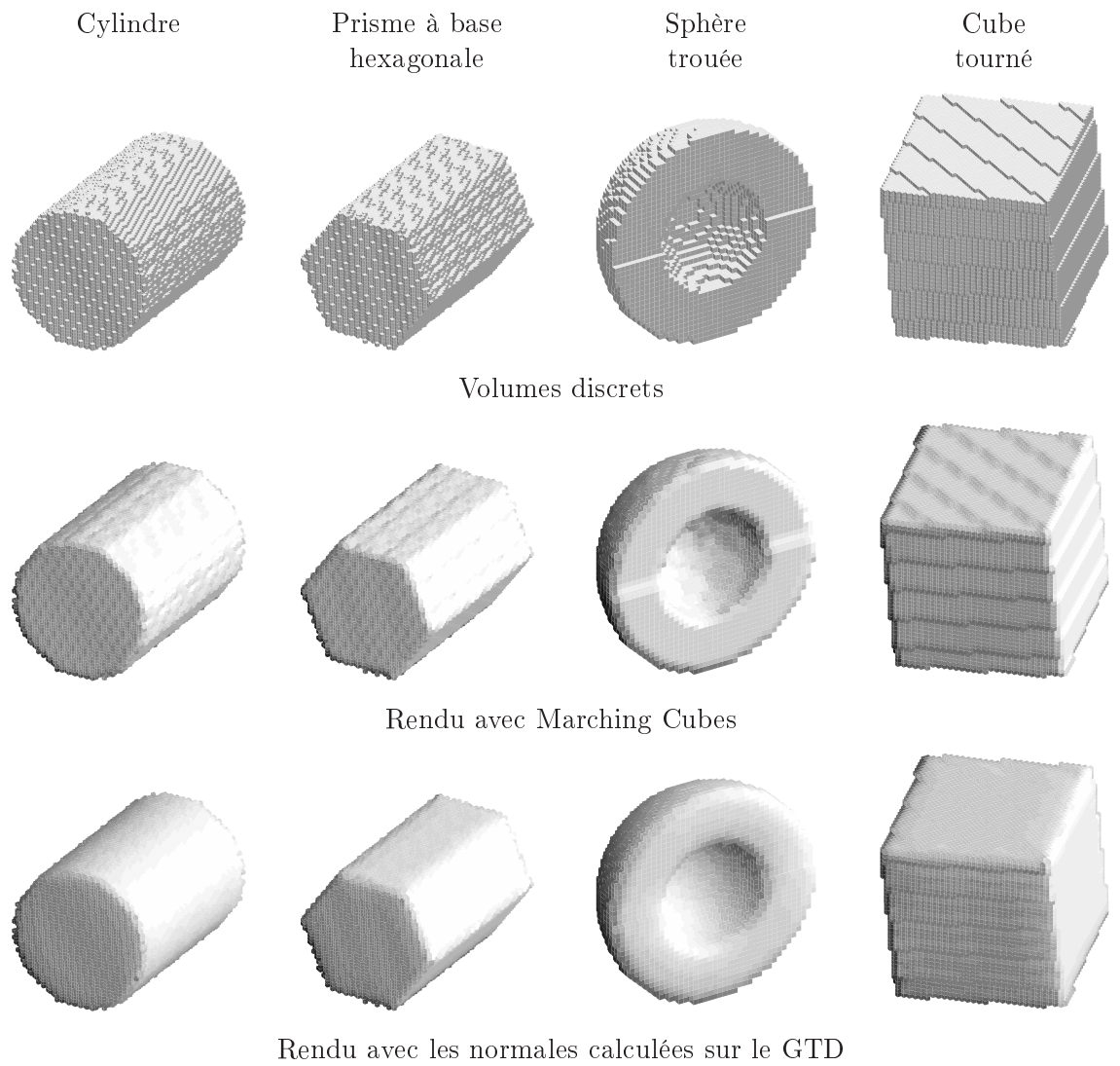


FIG. 5.6 –: *Comparaison entre les visualisations avec l’algorithme des Marching Cubes et avec les normales basées sur le GTD sur des formes géométriques simples.*

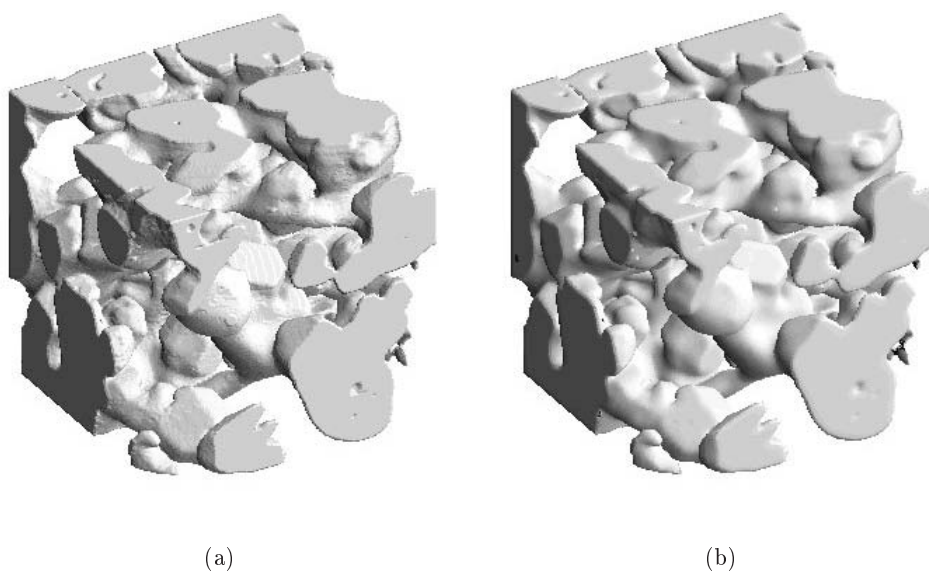


FIG. 5.7 – : Visualisation d'un échantillon de neige avec une estimation de normales basée sur l'algorithme des *Marching-Cubes* (a) et avec l'estimateur basé sur le *GTD* (b) (Images Météo-France CEN, ESRF, voir les explications dans le paragraphe A.3).

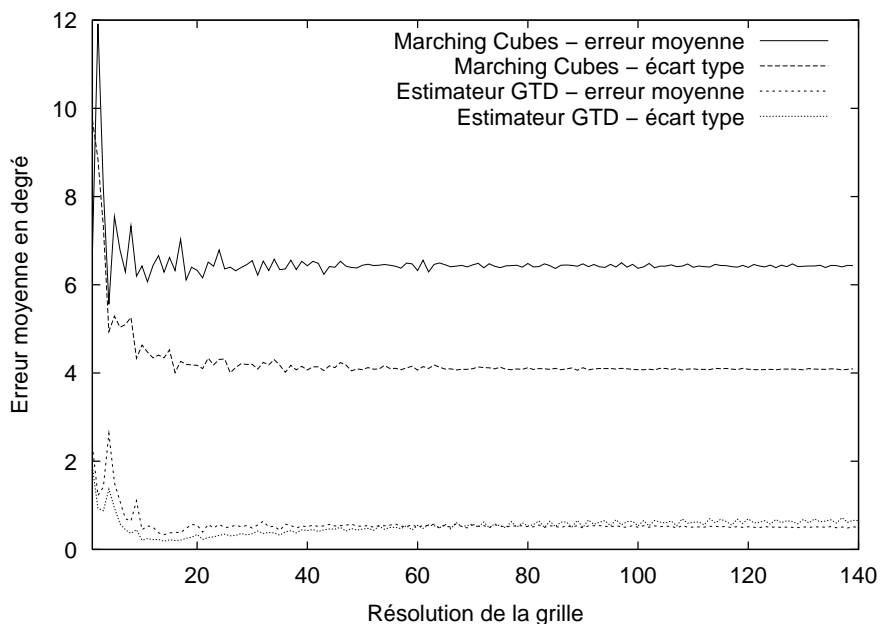
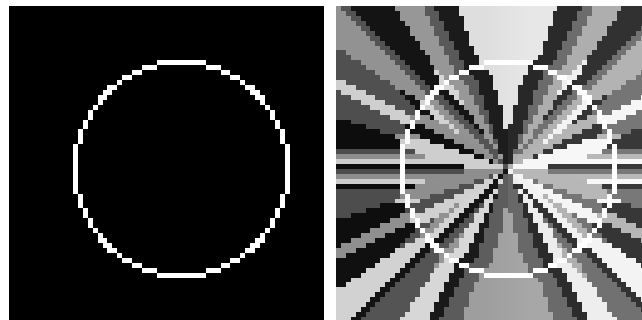
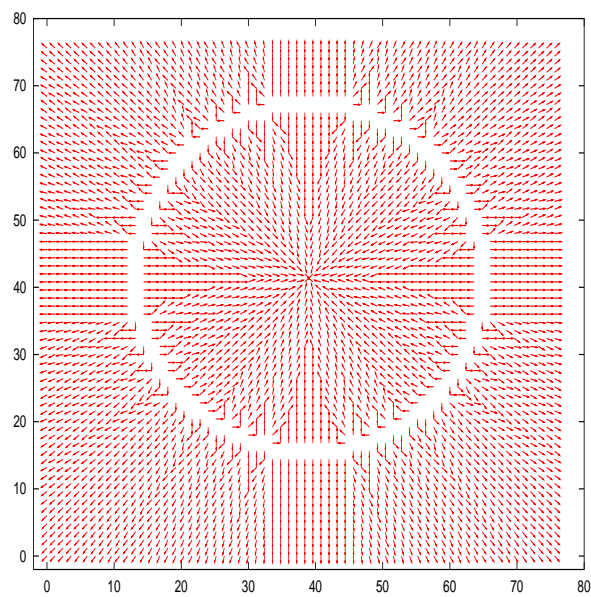


FIG. 5.8 – : Comparaison de l'estimation de la normale entre l'algorithme basé sur le *GTD* et l'algorithme issu des *Marching Cubes* sur des sphères de taille croissante.



(a)

(b)



(c)

FIG. 5.9 –: *Illustration 2D de l'utilisation du diagramme de Voronoï discret pour l'estimation directe du GTD : (a) contour d'un objet discret, (b) diagramme de Voronoï discret de ce contour et (c) association à chaque point d'un vecteur unitaire vers le point du contour le plus proche.*

5.3.2.2 Approches surfaciques

Parmi les approches surfaciques, une des premières solutions a été apportée par CHEN ET AL. (1985). Celle-ci se base sur une définition de la surface par complexes cellulaires. Dans ce cas, la surface discrète est composée de surfels munis d'une relation d'adjacence connue (voir paragraphe 1.3.1.2). Ainsi, CHEN ET AL. (1985) estiment la normale en un surfel s en regardant les différentes orientations des surfels adjacents à s (ceux partageant une arête avec s). Ils introduisent donc la notion de *normale élémentaire* qui correspond au vecteur unitaire orthogonal au surfel s , centré sur celui-ci et pointant vers l'extérieur de l'objet (voir figure 5.10). L'estimation de la normale se base donc sur la normale élémentaire en s ainsi que sur les quatre normales élémentaires des surfels adjacents à s . Cette normale estimée correspond à la moyenne de ces normales élémentaires (voir figure 5.11). Étant donné que nous n'avons que six orientations possibles de ces normales élémentaires, le nombre possible de normales estimées est assez faible.

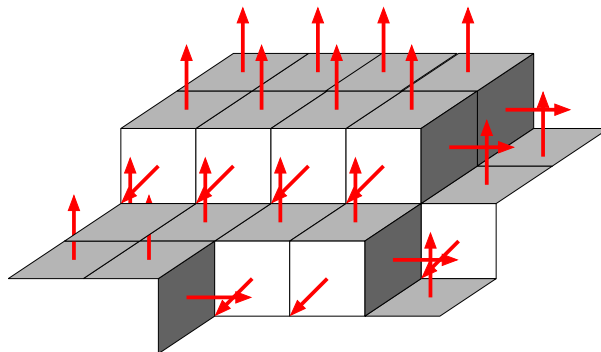


FIG. 5.10 – Illustration des normales élémentaires associées à un morceau de surface discrète.

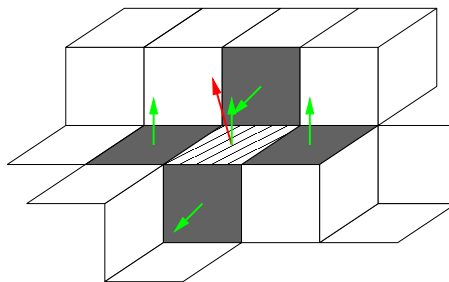


FIG. 5.11 – Approche de CHEN ET AL. (1985) pour l'estimation de la normale basée sur une décomposition cellulaire de la surface discrète.

Par la suite, PAPIER et FRANÇON (1998); PAPIER (1999) proposent l'utilisation d'un voisinage plus grand pour l'estimation de la normale. En se basant sur une définition topologique de ce voisinage (notion d'*ombrelle d'ordre n*), ils proposent un moyennage des normales élémentaires avec des pondérations de manière à favoriser l'apport des normales élémentaires proches de s dans le calcul de la moyenne.

Cette approche permet une bonne estimation des normales mais souffre d'un problème majeur introduit par le choix du voisinage. En effet, celui-ci est purement topologique et ne permet pas de contrôler l'*isotropie* du moyennage. Plus précisément, si nous considérons un plan parallèle à un axe de la grille, le voisinage topologique de PAPIER correspond à des rectangles concentriques (propagation en distance d_8).

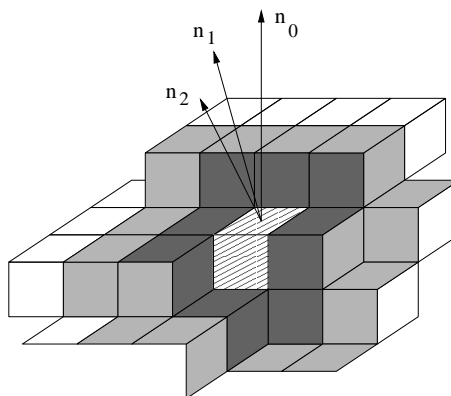


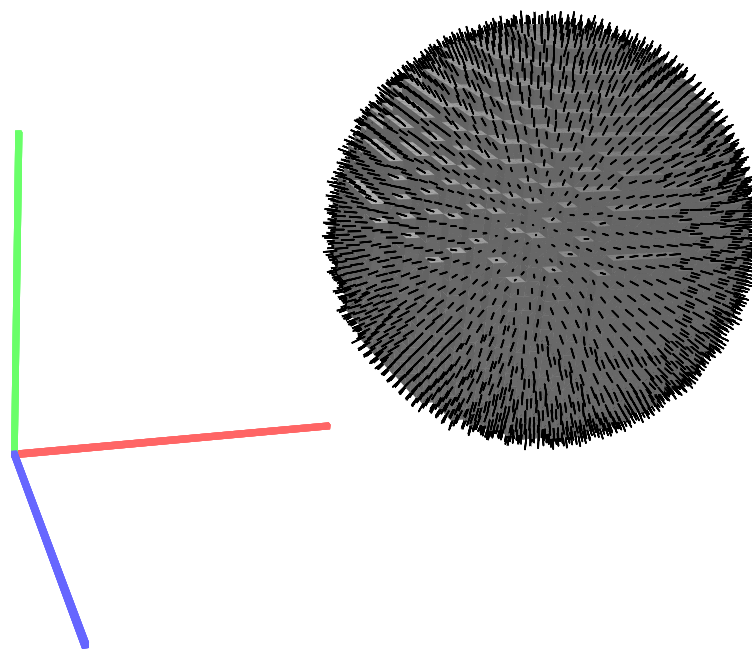
FIG. 5.12 – : *Approche de PAPIER (1999) pour l'estimation de la normale : l'estimation est construite en moyennant les normales élémentaires dans un certain voisinage. Des poids sont utilisés (inversement proportionnels à la distance au surfel s considéré) pour favoriser l'apport des normales élémentaires proches de s dans le calcul de la moyenne.*

Présentons maintenant une nouvelle version de l'algorithme de PAPIER (1999) dont le voisinage se base sur la métrique géodésique décrite dans le paragraphe 4.4.4. L'idée est de considérer tous les surfels qui sont à une distance géodésique inférieure à un certain seuil de s . Cette approche nous permet d'avoir un processus de pondération des normales élémentaires isotropique sur la surface. La figure 5.13 présente les résultats de ce calcul sur une sphère et sur un cube tourné. Les premières mesures d'erreur de cet estimateur indiquent une approximation de la normale un peu meilleure que l'approche de PAPIER (1999) mais de plus amples analyses expérimentales restent à effectuer.

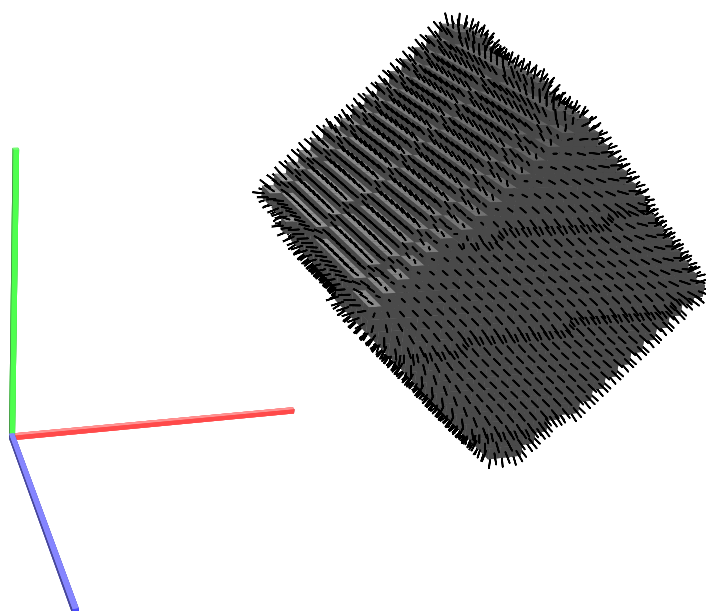
Le problème majeur avec ce genre d'algorithmes reste la construction du contexte qui ne correspond qu'à un voisinage de taille fixe autour du point considéré. Récemment, THÜRMER (2001) a proposé un calcul de contexte basé sur la reconnaissance de plan tangent discret. En fait, THÜRMER propose un algorithme d'estimation des normales en deux étapes : la première consiste en une estimation locale de celles-ci avec l'algorithme de THÜRMER et WÜTHRICH (1998) (voir ci-dessus) et la seconde consiste en un lissage de ces normales dans le contexte défini par le plan tangent discret. Or, comme nous l'avons vu précédemment, la reconnaissance de plan discret ou de plan tangent discret est un problème assez difficile sur le plan algorithmique. Nous voyons ici tout l'intérêt d'une estimation statistique de plan tangent discret comme il a été présenté dans le paragraphe 2.5. En effet, si par exemple nous pouvions stopper le processus de pondération des normales élémentaires par un test statistique d'appartenance à un plan discret, nous pourrions ainsi construire un estimateur très rapide et surtout adaptatif à la forme.

Nous terminons avec les approches basées sur une décomposition en coupes de l'objet discret. Ce type d'estimateur a été proposé initialement par LENOIR ET AL. (1996); LENOIR (1999) puis repris avec les notions de droites discrètes par TELLIER et DEBLED-RENNESON (1999).

Une coupe selon l'axe Oz de la grille d'indice k correspond à l'ensemble des voxels (x, y, k) de l'objet (avec $x, y \in \mathbb{Z}$). De la même manière, nous définissons les coupes selon l'axe Ox et selon l'axe Oy (voir illustration figure 5.14). Si nous considérons une surface discrète composée de surfels et les bords des différentes coupes 2D comme des ensembles de pointels et de lignels, nous avons la propriété importante que chaque



(a)



(b)

FIG. 5.13 – : *Exemple de calcul de normales basé sur une pondération de normales élémentaires avec un voisinage géodésique : (a) sur une sphère discrète et (b) sur un cube tourné.*

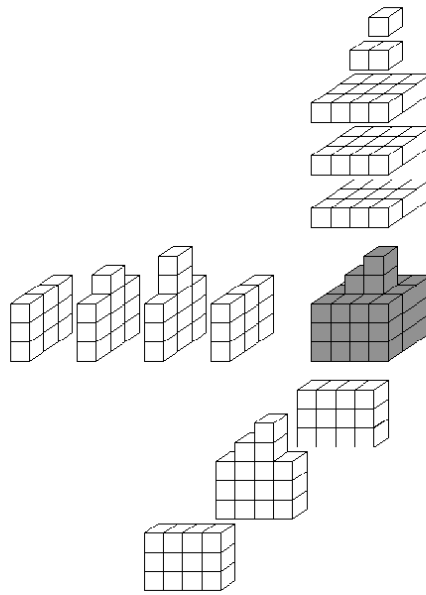


FIG. 5.14 – : *Décomposition en coupes d'un volume selon les axes de la grille.*

surfel engendre deux lignels dans deux coupes différentes (voir figure 5.15). Pour estimer la normale en un surfel s de la surface, LENOIR (1999) propose le calcul suivant : dans un premier temps, nous estimons les tangentes \vec{t}_1 et \vec{t}_2 en dimension 2 dans les deux coupes associées au surfel s . Ensuite, la normale en s est calculée par la formule :

$$\vec{n}(s) = \vec{t}_1(s) \times \vec{t}_2(s)$$

où ' \times ' représente le produit vectoriel usuel.

Ainsi, basé sur ce principe de décomposition du problème en deux problèmes 2D, LENOIR ET AL. (1996) proposent l'utilisation d'un filtre Gaussien récursif pour l'estimation des tangentes 2D et construisent les normales 3D grâce à la formule précédente. Dans une approche basée sur le calcul de tangentes discrètes, TELLIER et DEBLED-RENNESON (1999) proposent un algorithme similaire pour l'estimation de la normale 3D.

5.3.2.3 Estimateur optimal en temps avec convergence asymptotique

Dans ce paragraphe nous proposons une extension de ces approches par coupes en utilisant les résultats précédents sur le calcul de tangentes discrètes 2D et en analysant le comportement asymptotique de l'estimateur.

Tout d'abord, nous pouvons directement reprendre l'algorithme optimal de FESCHET ET TOUGNE (1999) pour le calcul des tangentes discrètes 2D. Ainsi, si nous notons n le nombre de surfels composant la surface discrète, le calcul des tangentes discrètes dans toutes les coupes se fait en temps linéaire en le nombre de surfels. Finalement, l'utilisation d'un tel algorithme nous permet l'écriture d'un algorithme d'estimation des normales discrètes 3D en $O(n)$ ce qui est optimal pour le problème.

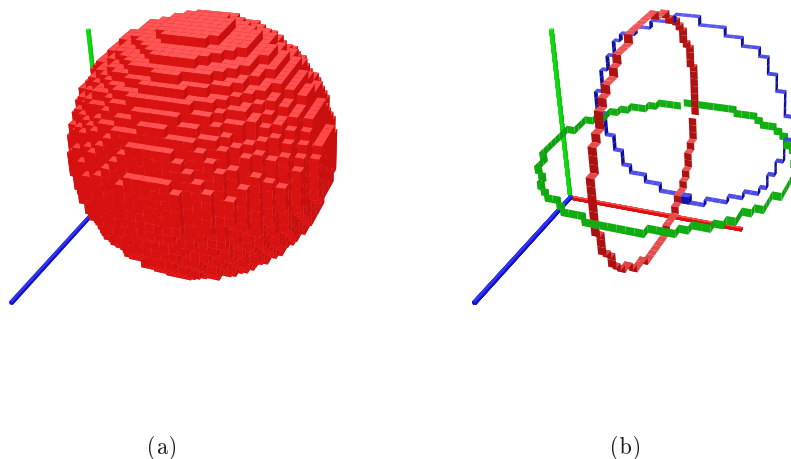


FIG. 5.15 –: Décomposition en coupes d'une surface discrète composée de surfels : (a) surface d'une sphère discrète et (b) exemples de coupes dans les trois axes de la grille.

De plus, nous avons le théorème suivant :

Théorème 5.2 *Soit \mathcal{S} une surface C^2 dans \mathbb{R}^3 à courbure bornée (avec quelques hypothèses techniques), l'estimation des normales surfaciques basée sur l'approche par coupes et utilisant les tangentes discrètes converge asymptotiquement.*

La preuve de ce théorème est présentée dans l'annexe B (théorème B.5). De manière informelle, si les normales convergent asymptotiquement dans chacune des coupes (voir théorème 5.1), le produit vectoriel aura aussi cette propriété de convergence asymptotique.

En conclusion, l'algorithme présenté ci-dessus est non seulement optimal en temps, mais a aussi la propriété de convergence asymptotique. Une perspective à ce travail consiste en l'implémentation et en l'évaluation de cet estimateur sur un jeu de tests significatifs pour des applications concrètes.

5.4 Longueur et aire

Lorsque l'on associe une *signature* à un contour discret (2D ou 3D), la normalisation des mesures présentes dans cette signature est primordiale avant toute comparaison de tels vecteurs ou plus généralement, avant toute utilisation de cette donnée dans un processus d'apprentissage.

Une normalisation classique des mesures se base sur un calcul préalable de la longueur ou l'aire du contour. Ainsi, si l'estimateur de longueur ou d'aire n'est stable ou n'est pas efficace, c'est toute la validité de la signature qui est remise en question.

5.4.1 Longueur d'une courbe discrète 2D

Dès les premières analyses de contours discrets (FREEMAN 1961 nous semble être un des premiers articles traitant du sujet), le calcul de longueur d'un contour discret est

apparu comme étant un descripteur de forme fondamental et des premières solutions furent proposées. Depuis, de nombreux algorithmes ont vu le jour pour estimer cette mesure euclidienne. Certaines de ces approches se basent sur une estimation de la métrique euclidienne par des approches locales (distances de chanfrein par exemple), d'autres considèrent une polygonalisation de la courbe discrète en utilisant par exemple des algorithmes de segmentation en droites discrètes ou de calcul du polygone interne de longueur minimale.

Nous présentons une évaluation comparative des estimateurs de longueur les plus récents. Cette évaluation porte, à la fois sur une analyse théorique basée sur la notion de convergence asymptotique, et sur une analyse expérimentale. Nous présentons aussi un estimateur de longueur basé sur la notion d'intégration de normales discrètes.

Cette partie a fait l'objet de l'article COEURJOLLY et KLETTE (2002).

5.4.1.1 Approches basées sur des métriques locales

Les estimateurs basés sur des métriques locales furent les premières solutions proposées (voir paragraphe 4.2.1). Rappelons que ces métriques locales définissent tout d'abord un ensemble de déplacements élémentaires et ensuite affectent des poids à ces déplacements. Par exemple, un estimateur simple consiste à affecter un poids de $1/r$ pour tout déplacement horizontal ou vertical sur une r -grille et $\sqrt{2}/r$ pour un déplacement diagonal. Le calcul des poids permettant l'estimation de la longueur se base sur le principe suivant : on considère une distribution probabiliste de segments de droite discrète et on calcule les poids de telle manière que l'erreur d'estimation de la longueur sur la distribution des segments soit minimale (DORST et SMEULDERS, 1987; VERWER, 1991).

Ainsi, en considérant les déplacements diagonaux et les déplacements parallèles aux axes de la grille, DORST et SMEULDERS (1987) définissent l'estimateur BLUE (Best Linear Unbiased Estimator) pour des segments discrets :

$$\mathcal{E}_{BLUE}(D_r(\gamma)) = \frac{1}{r} \cdot (0.948 \cdot n_i + 1.343 \cdot n_d) \quad (5.1)$$

où n_i représente le nombre de déplacements isothétiques (parallèles aux axes de la grille) et n_d le nombre de déplacements diagonaux du contour. Cet estimateur a un intérêt pratique très important du fait de sa simplicité. Cependant, celui-ci n'est pas convergent asymptotiquement dans le cas général mais a une convergence super-linéaire $O(r^{-1.5})$ dans le cas de segments de droite (DORST et SMEULDERS, 1991b). En d'autres termes, il ne converge asymptotiquement que pour la classe des segments de droite discrétisés.

Nous analysons aussi un estimateur proposé par VOSSEPOEL et SMEULDERS (1982), appelé *Corner count*. Celui-ci considère un déplacement élémentaire supplémentaire n_c qui correspond au nombre de changements de codes pair-impair dans le codage de FREEMAN.

$$\mathcal{E}_{CC}(D_r(\gamma)) = \frac{1}{r} \cdot (0.980 \cdot n_i + 1.406 \cdot n_d - 0.091 \cdot n_c)$$

Ce dernier estimateur n'est pas non plus convergent asymptotiquement dans le cas général.

5.4.1.2 Approches basées sur des approximations polygonales

Dans cette catégorie, la courbe discrète est approximée par une courbe polygonale. L'estimation de la longueur de la courbe discrète est ensuite donnée par la longueur de la courbe polygonale.

Dans le chapitre 2, nous nous sommes intéressés aux droites discrètes ainsi qu'aux algorithmes de segmentation d'une courbe discrète en segments discrets. Il est donc normal d'utiliser ces algorithmes pour construire une approximation polygonale.

Nous définissons donc deux estimateurs de longueur basés sur les droites discrètes : le premier, noté \mathcal{E}_{DR-DSS} , correspond à l'algorithme de segmentation de DEBLED-RENNESON et REVEILLÈS (1995) (voir paragraphe 2.2.2.5) sur des courbes de pixels. Le second, noté \mathcal{E}_{VK-DSS} , correspond à l'algorithme de segmentation de KOVALEVSKY (1990) sur une courbe discrète définie selon une approche complexe cellulaire (voir chapitre 1). La figure 5.16 illustre ces différentes approches.

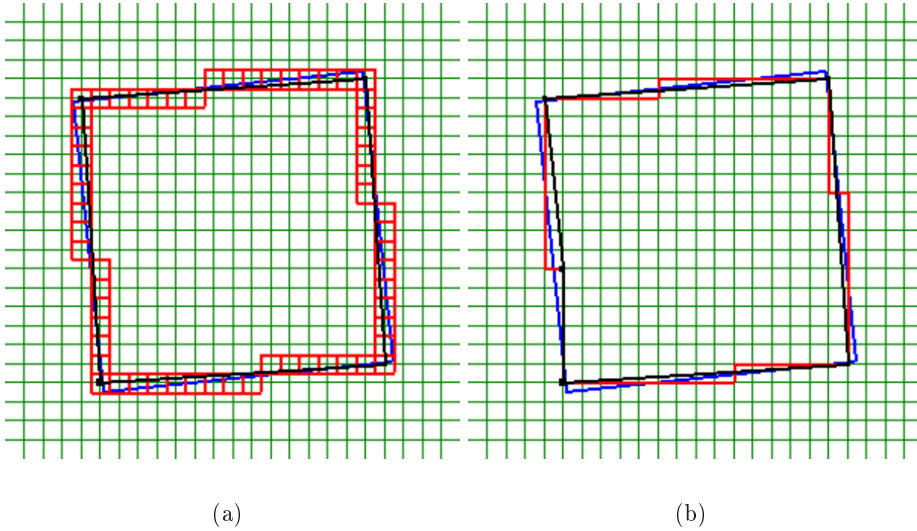


FIG. 5.16 – : Illustration des algorithmes de segmentation en droites discrètes : (a) \mathcal{E}_{DR-DSS} et (b) \mathcal{E}_{VK-DSS} sur un carré tourné.

Pour ces approches, nous pouvons formellement prouver la convergence asymptotique :

Théorème 5.3 (KLETTE et ŽUNIĆ (1999)) Soit P un objet polygonal convexe r -compact de \mathbb{R}^2 , il existe une résolution r_0 telle que pour tout $r \geq r_0$, toute polygonalisation par droite discrète de $D_r(\gamma)$ (avec γ le bord de P) est un polygone dont le périmètre p_r vérifie :

$$|\mathcal{L}(\gamma) - p_r| \leq \kappa \frac{2\pi}{r}$$

κ étant une constante dépendant de l'algorithme choisi.

Considérant une classe plus large de courbes euclidiennes (non nécessairement convexes), nous proposons un autre théorème :

Théorème 5.4 Soit Γ la famille des courbes dans \mathbb{R}^2 à courbure bornée (avec d'autres hypothèses techniques), l'estimateur de longueur basé sur une polygonalisation par droite discrète de $\gamma \in \Gamma$ a la propriété de convergence asymptotique.

La preuve de ce théorème, présentée dans COEURJOLLY ET AL. (2001a), est donnée en annexe B (théorème B.2).

Une autre approximation polygonale, qui a été proposée par SLOBODA ET ZATKO (1996), consiste en un calcul de polygone de longueur minimale (MLP pour *minimum length polygon*) dans une région englobant la courbe réelle γ .

Plus formellement, nous considérons la courbe discrète $J_r^+(\gamma)$ (resp. $J_r^-(\gamma)$) qui correspond au bord de Jordan interne (resp. externe) à γ (similaire à une discrétisation OBQ, resp. BBQ, de γ). Ainsi, l'approximation polygonale de γ consiste en le calcul du polygone de longueur minimale compris dans l'anneau défini par $J_r^+(\gamma)/J_r^-(\gamma)$ (voir illustration figure 5.17). Ce polygone est unique et nous permet de définir l'estimateur de longueur \mathcal{E}_{SZ-MLP} .

Nous utilisons l'algorithme de calcul de MLP proposé par KLETTE ET AL. (1999) dont la complexité est linéaire en la taille de la courbe discrète.

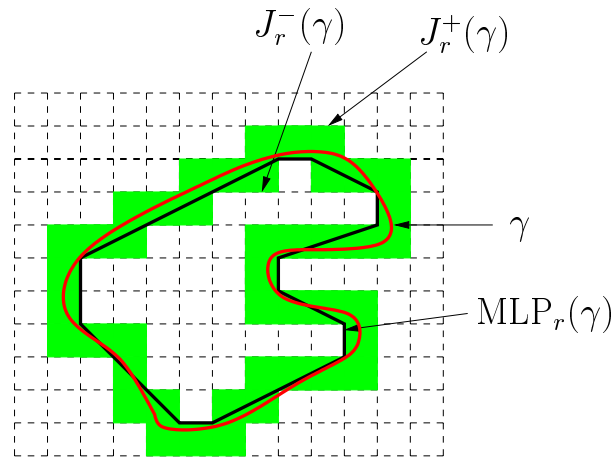


FIG. 5.17 – Notations pour le calcul de polygone de longueur minimale (MLP)

Cet estimateur converge asymptotiquement :

Théorème 5.5 (SLOBODA ET AL. (1998)) *Soit P un objet polygonal convexe r -compact de \mathbb{R}^2 , il existe une résolution r_0 telle que pour tout $r \geq r_0$, l'estimateur de longueur basé sur SZ-MLP converge asymptotiquement.*

La figure 5.18-(a) illustre cet algorithme sur un carré tourné.

Finalement, nous présentons une dernière approximation basée sur un calcul de MLP mais dans un anneau un peu différent. Dans cette approche proposée par ASANO ET AL. (2001), il est introduit un paramètre réel δ tel que $0 < \delta < \frac{1}{2r}$. Ainsi deux nouvelles courbes de Jordan dans lesquelles chaque sommet v_i peut être déplacé de δ sont définies (voir figure 5.19). Ce processus d'ajustement se fait en deux parcours des sommets de chaque bord. Enfin, un calcul de MLP classique est appliqué dans cet anneau. Notons que cette approche n'est pas entièrement discrète puisque les sommets de la polygonalisation ne sont pas des points de la grille. La figure 5.18-(b) présente un exemple de calcul de cette approximation. Dans ce qui suit, nous notons \mathcal{E}_{AS-MLP} l'estimateur de longueur basé sur cette approximation polygonale.

Cette technique, qui effectue un *lissage* des courbes de Jordan et donc du MLP, est assez similaire à la définition du chemin euclidien proposé par VIALARD (1996a).

Une fois encore, nous avons le théorème suivant :

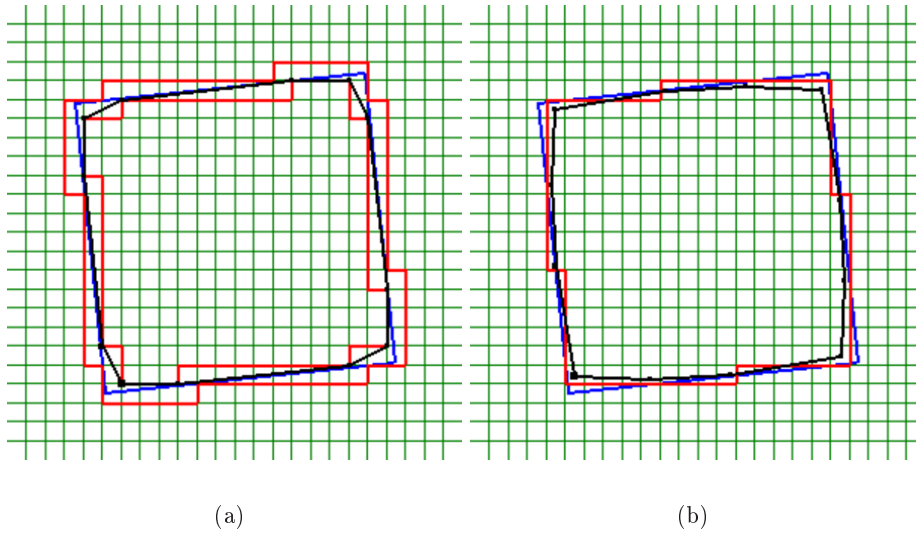


FIG. 5.18 –: Illustration des algorithmes SZ-MLP et AS-MLP : (a) algorithme SZ-MLP avec les courbes de Jordan $J_r^+(\gamma)$ et $J_r^-(\gamma)$, le polygone solution est celui en noir. (b) Approximation basée sur l'algorithme AS-MLP avec $\delta = \frac{1}{4}$. Remarquons que les sommets de la courbe polygonale noire ne sont plus discrets.

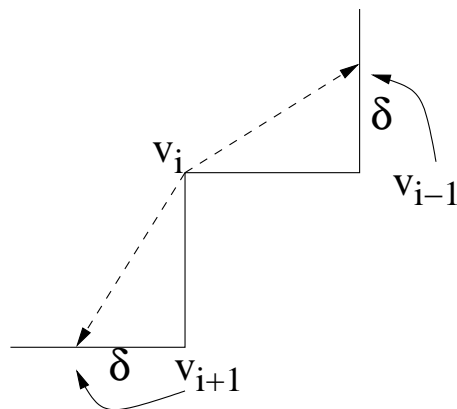


FIG. 5.19 –: Approche AS-MLP : les sommets des bords de Jordan sont déplacés en fonction d'un paramètre δ .

Théorème 5.6 (ASANO ET AL. (2001)) *Soit P un objet polygonal convexe r -compact de \mathbb{R}^2 , il existe une résolution r_0 telle que pour tout $r \geq r_0$, l'estimateur de longueur basé sur AS-MLP converge asymptotiquement.*

Cet algorithme a aussi une complexité linéaire en la taille de la courbe discrète considérée.

5.4.1.3 Approches basées sur l'intégration des normales discrètes

Nous proposons une approche basée sur l'intégration d'un champ de normales discrètes. L'objectif étant de se ramener à une version discrète de la formule classique en géométrie euclidienne :

$$\mathcal{L}(\gamma) = \int_{\gamma} \vec{n}(s) ds \quad (5.2)$$

Cette idée de calcul de longueur par intégration *discrète* est assez ancienne. En effet, elle fut proposée pour la première fois par ELLIS ET AL. (1979) et par LENOIR (1999) pour le calcul de l'aire d'une surface discrète.

L'idée est de remplacer l'intégrale par une somme finie, le vecteur normal par une estimation de celui-ci, notée $\vec{n}^*(s)$, et le terme ds par un produit scalaire avec un vecteur élémentaire \vec{n}_{el} . Dans une représentation cellulaire du contour (séquence de pointels et de lignels), le vecteur élémentaire en un lignal s est un vecteur unitaire orthogonal à s pointant vers l'intérieur de la forme (voir figure 5.20).

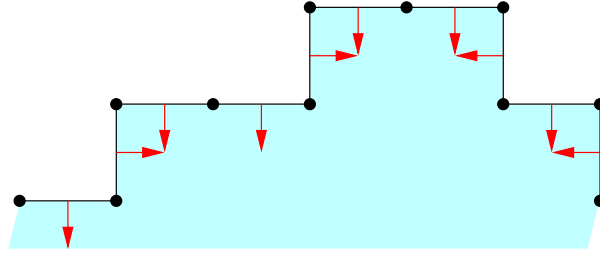


FIG. 5.20 – Illustration des vecteurs élémentaires à une courbe discrète.

Finalement, nous obtenons l'estimateur de longueur donné par la formule :

$$\mathcal{E}_{TAN}(D_r(\gamma)) = \sum_{s \in D_r(\gamma)} \vec{n}^*(s) \cdot \vec{n}_{el}(s)$$

Ce processus de sommation de produits scalaires entre ces deux vecteurs est appelé *processus d'intégration discret*. L'idée intuitive de ce processus est de considérer l'apport de chaque lignal à l'estimation de longueur de la courbe.

Nous avons montré le théorème suivant :

Théorème 5.7 *Soit Γ la famille des courbes C^2 dans \mathbb{R}^2 à courbure bornée, l'estimateur de longueur basée sur le processus d'intégration discret des normales converge asymptotiquement si et seulement si l'estimation des normales a cette propriété de convergence.*

La preuve de ce théorème est présentée dans l'annexe B (corollaire B.4). Dans cette annexe, nous prouvons en fait le problème plus général du calcul d'aire d'une surface discrète (voir paragraphe 5.4.3). Ce dernier problème se ramène, sans difficulté, au problème de l'estimation de longueur d'une courbe discrète.

Or si nous utilisons l'estimateur de normale discrète 2D présenté précédemment, nous obtenons le corollaire suivant :

Corollaire 5.1 *Soit Γ la famille des courbes C^2 à courbure bornée (plus quelques hypothèses techniques), l'estimateur de longueur \mathcal{E}_{TAN} basé sur les normales discrètes du paragraphe 5.3 converge asymptotiquement.*

Preuve : *Il suffit simplement de considérer les théorèmes 5.7 et 5.1. Les hypothèses techniques sont discutées dans l'annexe B. \square*

Dans le cas où la courbe discrète manipulée est donnée sous forme de séquence de pixels connectés, nous pouvons proposer un estimateur de longueur ayant les mêmes propriétés. Dans ce cas, l'estimation de la normale \vec{n}^* est donnée en chaque pixel p et l'apport de ce pixel pour l'estimation de longueur est donnée par :

$$a(p) = \frac{\|\vec{n}^*(p)\|}{\max(|n_x^*(p)|, |n_y^*(p)|)}$$

De manière intuitive, cette mesure quantifie l'écart entre le vecteur normal et l'axe de la grille le plus proche.

Cet apport est donné par la longueur de l'intersection entre la tangente au point p et le voxel considéré (voir figure 5.21). Ainsi, pour ce type de courbe nous avons l'estimateur :

$$\mathcal{E}_{TAN_PIX}(D_r(\gamma)) = \sum_{p \in D_r(\gamma)} a(p)$$

Ce dernier a exactement les mêmes propriétés de convergence asymptotique que \mathcal{E}_{TAN} .

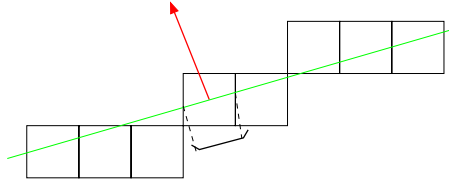


FIG. 5.21 – : *Calcul de l'apport d'un pixel dans l'estimation de longueur par intersection entre le pixel et la tangente estimée en celui-ci.*

La figure 5.22 illustre les deux estimateurs précédents.

En utilisant l'algorithme optimal en temps de FESCHET et TOUGNE (1999), les algorithmes implémentant les estimateurs \mathcal{E}_{TAN} et \mathcal{E}_{TAN_PIX} ont une complexité linéaire en la taille de la courbe discrète.

5.4.1.4 Évaluation comparative

Nous présentons maintenant une évaluation comparative des estimateurs de longueurs présentés. Cette évaluation s'effectue, sur le plan théorique avec notamment la propriété de convergence asymptotique, et sur le plan expérimental avec des analyses comportementales des différentes méthodes sur des données réelles.

D'un point de vue théorique, nous évaluons les estimateurs sur les propriétés suivantes :

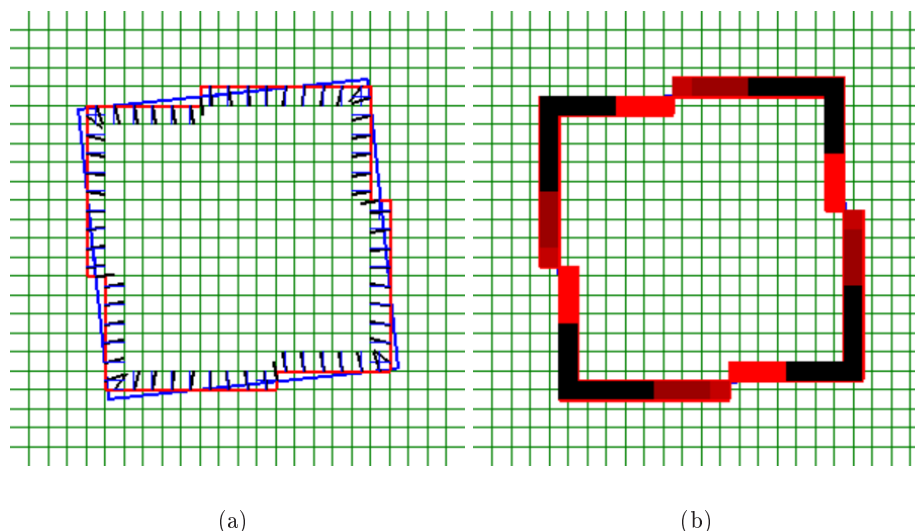


FIG. 5.22 — Illustration des estimateurs de longueur basés sur l'intégration des normales discrètes : (a) estimateur \mathcal{E}_{TAN} , à chaque ligne est associé deux vecteurs, l'estimation de la normale et la normale élémentaire ; et (b) estimateur \mathcal{E}_{TAN_PIX} , dans ce cas, l'intensité associée à chaque pixel correspond à la mesure $a(p)$ (le dégradé du clair au sombre correspond aux valeurs croissantes de $a(p)$).

Convergence asymptotique : est-ce que la mesure donnée par l'estimateur converge asymptotiquement vers la mesure euclidienne de l'objet continu sous-jacent à la discrétisation ?

Unicité : étant donné une courbe discrète, est-ce que l'estimation est unique ?

Calcul discret : est-ce que l'algorithme implémentant l'estimateur considéré est purement discret ? Plus précisément, est-ce que le calcul peut se faire en nombres entiers ?

Extension en 3D : est-il possible d'étendre l'estimateur aux courbes discrètes tridimensionnelles ?

Cette dernière propriété sera reprise dans le paragraphe suivant.

Le tableau 5.1 illustre le comportement de tous les estimateurs présentés vis-à-vis des propriétés ci-dessus. Rappelons cependant que tous les algorithmes ont une complexité linéaire en le nombre de pixels.

Sur le plan expérimental, nous considérons les formes géométriques multi-grilles présentées dans la figure 5.2 et plus précisément leurs discrétisées sur des grilles de plus en plus fines. De plus, pour une résolution donnée, un certain nombre de rotations des formes sont prises en compte. Finalement, nous évaluons tous les estimateurs sur l'ensemble des courbes discrètes obtenues.

Lors de l'évaluation des estimateurs, deux mesures sont analysées. La première consiste en un calcul d'erreur relative en pourcentage par rapport à la longueur euclidienne exacte des formes. Cette mesure nous permet d'évaluer expérimentalement la propriété de convergence asymptotique. La seconde mesure utilisée n'est définie que sur les approches polygonales. Celle-ci est donnée par le produit entre l'erreur relative de l'estimateur et le nombre d'arêtes de l'approximation polygonale. Ce calcul proposé par KLETTE ET AL. (1999) permet d'évaluer l'efficacité de l'approche polygonale pour l'estimation de longueur.

Approche	Convergence asymptotique	Unicité	Discret	Extension 3D	Références principales
Métriques locales	non	oui	cela dépend	BECKERS et SMEULDERS (1992); JONAS et KIRYATI (1998)	DORST et SMEULDERS (1987); VERWER (1991)
DR-DSS et VK-DSS	oui	non	oui	COEURJOLLY ET AL. (2001a)	DEBLED-RENNESON et REVEILLÈS (1995); KOVALEVSKY (1990)
SZ-MLP	oui	oui	oui	KLETTE et BÜLOW (2000)	SLOBODA et ZATKO (1996); KLETTE ET AL. (1999)
AS-MLP	oui	oui pour δ fixé	non	-	ASANO ET AL. (2001)
TAN et TAN_PIX	oui	oui	oui	-	ELLIS ET AL. (1979); COEURJOLLY et KLETTE (2002)

TAB. 5.1 –: *Évaluation théorique des différents estimateurs et principales références bibliographiques.*

Dans notre analyse, nous avons considéré un processus de discrétisation des formes de la figure 5.2 sur une grille de résolution comprise entre 30 et 1000 pixels. De plus pour chaque résolution, les rotations d'angles compris entre 0 et 90 degrés par pas de 1 sont considérées. Finalement, les mesures associées à un estimateur correspondent à des moyennes des mesures obtenues sur toutes les formes et pour toutes les rotations de celles-ci. De plus, un moyennage sur une fenêtre de taille 16 permet un lissage des résultats et donc une meilleure lisibilité des courbes.

La figure 5.23 présente l'analyse de convergence asymptotique expérimentale basée sur l'erreur relative. Comme nous l'avons vu dans l'analyse théorique, toutes les mesures d'erreur des estimateurs, sauf pour BLUE et CC, convergent vers 0 pour une résolution croissante. Les mesures d'erreur des estimateurs basés sur des métriques locales (BLUE et CC) semblent se stabiliser vers une erreur de 2%. En détaillant un peu plus les résultats, nous pouvons observer que les meilleurs résultats sont obtenus par les algorithmes de segmentation en droites discrètes, par l'algorithme AS-MLP ainsi que par l'algorithme TAN. Ensuite nous avons l'approche SZ-MLP et enfin les estimateurs locaux.

La figure 5.24 présente la mesure d'efficacité de l'approximation polygonale pour les algorithmes VK-DSS, DR-DSS, SZ-MLP et AS-MLP. Une valeur faible de cette mesure indique un bon compromis entre l'estimation de longueur et la qualité, en nombre d'arêtes, de l'approximation polygonale. Les meilleurs résultats sont obtenus par les approches basées sur les droites discrètes.

Afin de mieux analyser le caractère anisotropique des estimateurs, nous avons considéré l'estimation de longueur de différentes rotations d'un carré sur une grille de taille fixe (voir figure 5.25). Comme nous pouvions nous y attendre, les estimateurs basés sur des métriques locales ont de très mauvais comportements pour ce type de tests. En effet, certaines orientations de la grille sont privilégiées (45° pour BLUE et 0° et 90° pour CC).

Enfin, la figure 5.26 montre les différents temps de calcul des estimateurs. Nous pouvons alors remarquer que la différence entre les estimateurs locaux et les algorithmes utilisant des outils de la géométrie discrète comme les droites discrètes est faible.

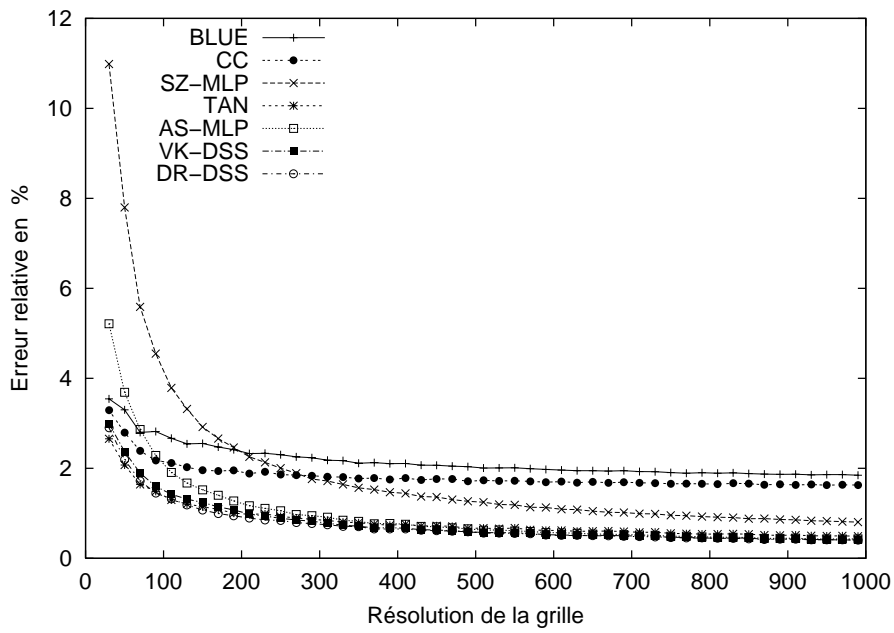
En conclusion, les estimateurs basés sur des métriques locales ne se justifient, ni sur le plan théorique, ni sur le plan expérimental et même leur rapidité peut être discutable dans des cas pratiques.

5.4.1.5 Remarques sur la plate-forme de tests

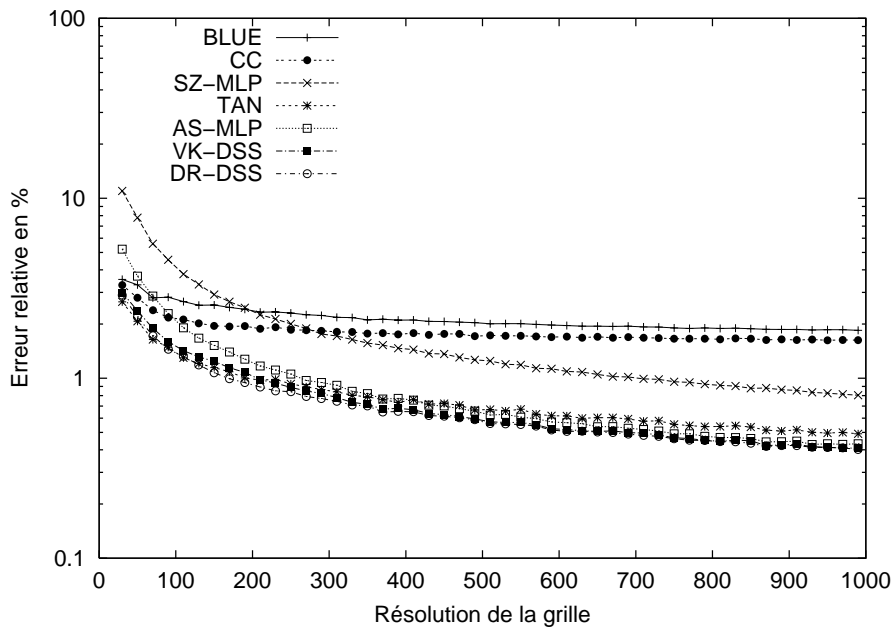
Toutes les analyses expérimentales présentées dans le paragraphe précédent ont été obtenues sur une plate-forme de tests implémentant tous les algorithmes sous une structure de classes C++.

Cette plate-forme contient :

- une série de classes implémentant les différentes formes multi-grilles de la figure 5.2 en fonction de l'orientation de la grille et la résolution de cette dernière ;
- une série de classes implémentant les différents algorithmes ;
- un système d'évaluation d'un estimateur pour une courbe donnée basé sur la lecture de scripts, ce qui permet d'effectuer les tests de convergence asymptotique très facilement.



(a)



(b)

FIG. 5.23 — Convergence asymptotique expérimentale des estimateurs de longueur : (a) erreur relative en pourcent de chaque estimateur sur le jeu de test et (b) mêmes résultats mais sur une échelle logarithmique en y de manière à faciliter la lecture des courbes pour les valeurs proches de 0.

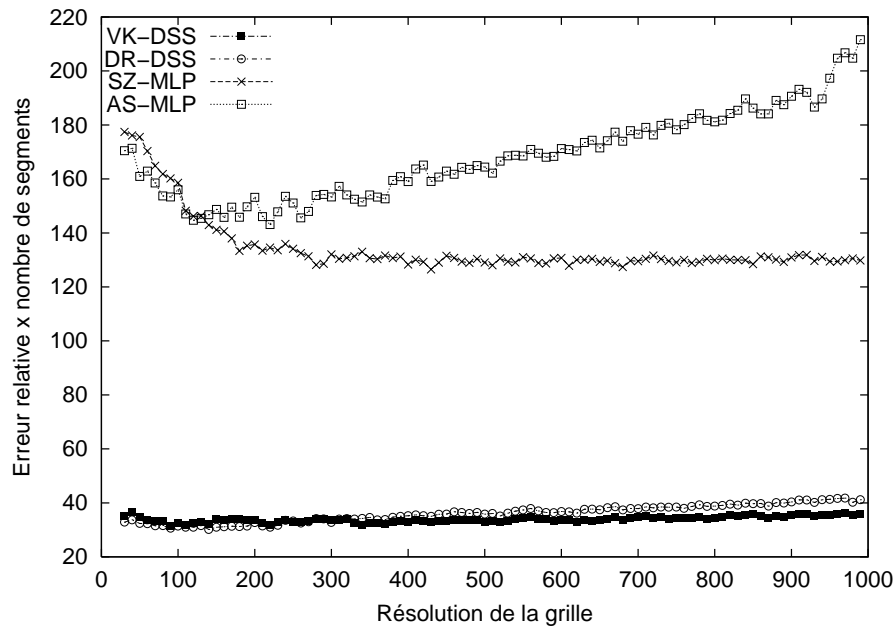


FIG. 5.24 — : *Mesure d'efficacité de l'approximation polygonale pour l'estimation de longueur : en abscisse se trouve la résolution de la grille et en ordonnée le produit entre le nombre d'arêtes de l'approximation polygonale et l'erreur relative en pourcent.*

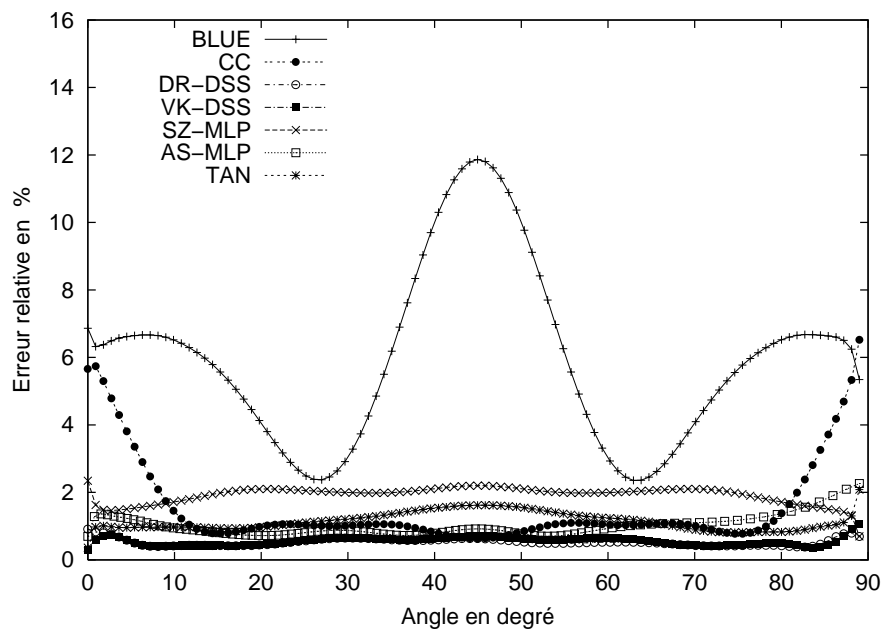


FIG. 5.25 — : *Erreur relative sur un carré de taille fixe sur lequel une rotation est effectuée : en abscisse se trouve l'angle de rotation entre 0 et 90 degrés et en ordonnée, l'erreur relative des différents estimateurs.*

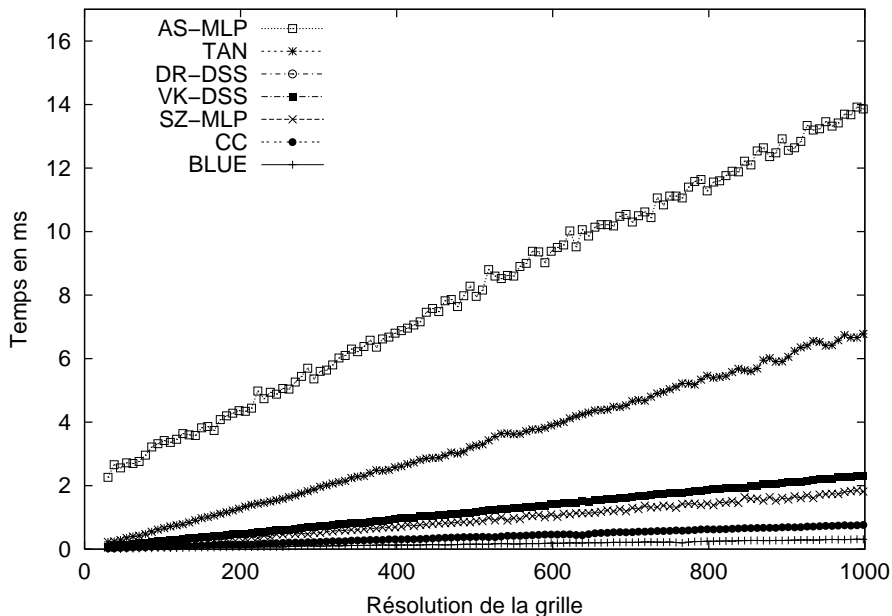


FIG. 5.26 —: Temps de calcul des différents estimateurs sur une station Sparc Ultra 10.

Toutes ces classes sont disponibles de manière libre sur le site *Digital Geometry and Topology*² maintenu par R. KLETTE.

Ce projet a démarré en 1999 à l'initiative de BEN YIP et REINHARD KLETTE du laboratoire CITR de l'université de Nouvelle-Zélande avec la création de la structure globale et l'implémentation des algorithmes SZ-MLP et VK-DSS. Par la suite, TETSUO ASANO and YASUYUKI KAWAMURA du laboratoire JAIST au Japon ont ajouté l'algorithme AS-MLP.

Enfin, nous avons implémenté les algorithmes BLUE, CC, DR-DSS, TAN et TAN_PIX, ainsi que les scripts permettant une analyse multi-grille efficace.

5.4.2 Longueur d'une courbe discrète 3D

Nous nous intéressons maintenant à l'estimation de la longueur de courbes discrètes tridimensionnelles. Dans ce cas, nous considérons une courbe $\gamma : [0, 1] \rightarrow \mathbb{R}^3$ et un processus de discrétisation multi-grille $D_r(\gamma)$ adapté.

Comme nous l'avons présenté dans le tableau 5.1, des extensions des algorithmes précédents existent pour l'estimation de longueur de courbes discrètes 3D. Ainsi, une première approche très simple consiste en une généralisation 3D des estimateurs basés sur des métriques locales. Dans ce cas, les déplacements élémentaires considérés sont par exemple les 6-adjacences de voxels (notées n_a), les adjacences de voxels par lignels (notées n_b) et les adjacences de voxels par pointels (notées n_c). Les poids associés à ces déplacements sont calculés de manière à minimiser l'erreur de l'estimation de longueur pour une distribution aléatoire de segments de droites réelles. Par exemple, BECKERS et SMEULDERS (1992) considèrent l'estimateur :

$$\mathcal{E}_{Beckers}(D_r(\gamma)) = 0.8875n_a + 1.34224n_b + 1.59772n_c$$

2. <http://http://www.citr.auckland.ac.nz/dgt>, un miroir est disponible à l'adresse <http://eric.univ-lyon2.fr/~dcoeurjo/download.html>

Par la suite, JONAS et KIRYATI (1998) proposent des estimateurs basés sur d'autres déplacements élémentaires.

En considérant des courbes discrètes 3D 6-connexes (*cube-curve*), KLETTE et BÜLOW (2000) proposent une approximation polygonale basée sur le polygone de longueur minimale inscrit dans ce volume (voir figure 5.27). Ce polygone 3D se calcule en deux étapes :

- dans un premier temps, des sommets *candidats* sont initialisés sur des arêtes dites *critiques* (lignels appartenant à trois voxels adjacents) ;
- des optimisations locales (suppression de sommets, déplacement le long des arêtes...) sont effectuées afin de réduire la longueur du polygone.

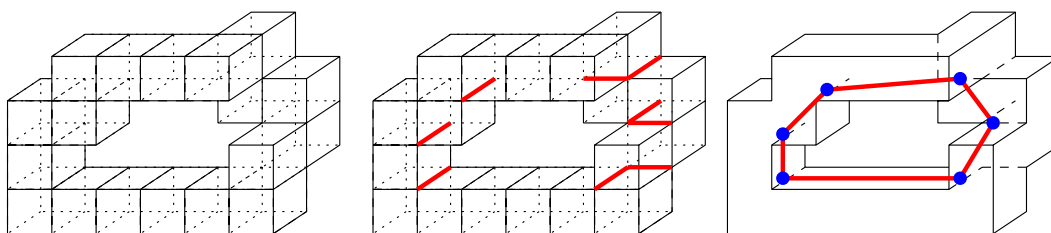


FIG. 5.27 —: Illustration du calcul du polygone de longueur minimale associé à une courbe discrète 6-connexe : courbe discrète initiale, sélection des arêtes critiques et calcul du polygone de longueur minimale.

Cet estimateur, noté \mathcal{E}_{3D-MLP} permet l'estimation de longueur d'une courbe 3D mais de nombreux problèmes existent sur le plan algorithmique. En effet, la phase d'optimisation de l'algorithme (seconde étape) est appliquée jusqu'à ce qu'aucune modification ne réduise la longueur du polygone MLP. Or, étant donné que les sommets de ce polygone peuvent se déplacer sur les arêtes critiques, non seulement ce calcul n'est plus discret mais en plus, l'algorithme permettant de trouver le minimum est exponentiel (l'algorithme est NP-complet) en temps. Ainsi, KLETTE et BÜLOW (2000) ont introduit un τ -MLP qui correspond au polygone dont la distance diffère de l'optimal de τ . Enfin, les auteurs proposent un algorithme non-linéaire pour calculer un tel polygone.

Sur le plan de la convergence asymptotique, les problèmes de définition et de construction du polygone MLP en 3D ne permettent pas d'avoir de preuve de convergence.

Dans COEURJOLLY ET AL. (2001a), nous avons proposé un algorithme de calcul de longueur basé sur la segmentation de la courbe en droites discrètes 3D (voir algorithme 10). Ainsi, cet estimateur se base sur l'approximation polygonale en droites discrètes pour estimer la longueur de la courbe discrète (voir figure 2.25 pour quelques exemples de segmentation). Rappelons que cet algorithme est totalement discret et que le calcul de la segmentation s'effectue en $O(n)$ où n est le nombre de voxels de la courbe discrète, ce qui est optimal pour le problème.

D'un point de vue théorique, nous avons le théorème suivant :

Théorème 5.8 *Soit Γ la famille des courbes dans \mathbb{R}^3 à courbure bornée (avec d'autres hypothèses techniques), l'estimateur de longueur basé sur une polygonalisation en droites discrètes 3D de la discrétisation de $\gamma \in \Gamma$ a la propriété de convergence asymptotique.*

Dans la preuve donnée dans l'annexe B (théorème B.2), nous avons utilisé un schéma général de preuve basé sur le fait qu'un segment de droite discrète faisant parti de la segmentation de la courbe $D_r(\gamma)$ peut être vu comme un *tube* dont l'épaisseur est contrôlée et qui englobe un morceau de γ . Ce schéma est valide quelle que soit la dimension du problème et c'est pour cela que nous pouvons utiliser exactement la même preuve pour le cas 2D que pour le cas 3D.

Finalement, nous avons un estimateur de longueur de courbes discrètes 3D, noté \mathcal{E}_{3D-DSS} , dont le calcul est optimal en temps, purement discret et qui converge asymptotiquement.

D'un point de vue expérimental, il est assez difficile de proposer un jeu de tests très pertinent en 3D. Dans les analyses expérimentales des figures 5.28 et 5.29, nous avons considéré des cercles et des ellipses 3D. Dans la figure 5.29-(b) nous avons tracé une mesure que nous n'avions pas présentée dans le paragraphe précédent, celle-ci est l'inverse de l'erreur relative en pourcentage. Cette mesure nous permet de nous rendre compte de la vitesse de convergence de l'estimation. Dans notre cas, la courbe se stabilise vers une vitesse de convergence linéaire.

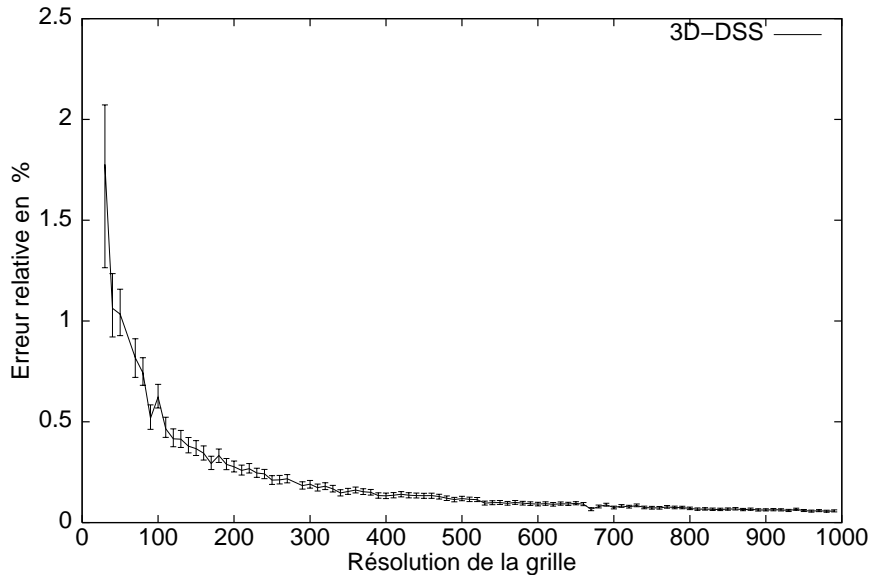
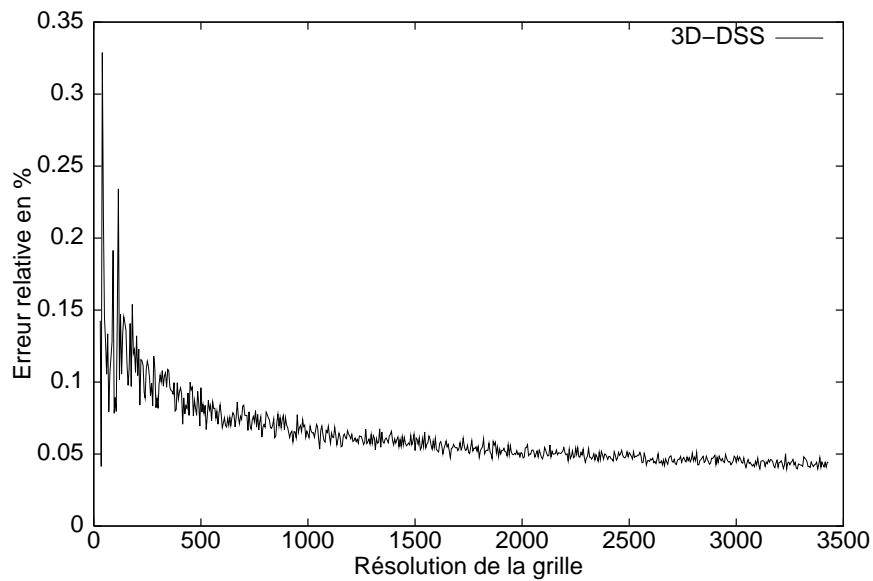
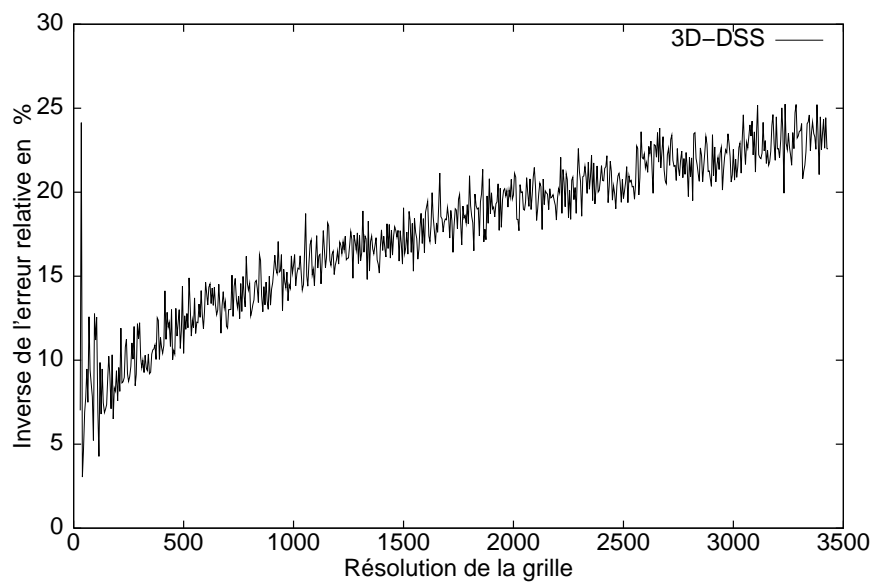


FIG. 5.28 — : *Analyse expérimentale de convergence asymptotique de l'estimateur \mathcal{E}_{3D-DSS} : convergence sur un cercle 3D. Les barres d'erreur correspondent aux différentes mesures observées en fonction du point de départ considéré de la segmentation.*

Une perspective importante de ce travail sur les longueurs de courbes discrètes 3D est bien évidemment la mise en place d'une plate-forme de test similaire à celle utilisée dans le cas 2D. Un travail préalable serait la désignation d'une famille de courbes réelles 3D sur lesquelles nous avons la mesure exacte de longueur et que nous discrétiserons en multi-grille.



(a)



(b)

FIG. 5.29 – : *Analyse expérimentale de convergence asymptotique de l'estimateur \mathcal{E}_{3D-DSS} : (a) sur une ellipse 3D dont le rapport entre les longueurs d'axe est $\frac{1}{\sqrt{2}}$ et (b) inverse de l'erreur relative sur les ellipses 3D afin d'observer la vitesse de convergence.*

5.4.3 Aire d'une surface discrète

En dimension 3, le calcul de l'aire de la surface d'un objet discret est un descripteur de forme très important. En effet, de la même manière qu'en 2D, cette mesure permet une normalisation de toutes les autres caractéristiques que nous pourrions extraire de ces surfaces. D'un point de vue pratique, cette mesure nous permet d'extraire une information physique très importante dans le cas d'analyses d'échantillons de neige (voir paragraphe A.3).

Nous allons, ici aussi, nous intéresser à la notion de convergence asymptotique. Pour cela, nous reformulons la propriété de convergence asymptotique dans le cas des surfaces dans \mathbb{R}^3 . Nous notons \mathcal{S} une surface de JORDAN dans \mathbb{R}^3 . Nous considérons un processus de discrétisation, noté D_r sur une grille 3D de résolution r . Soit $\mathcal{A}(\mathcal{S})$ l'aire de la surface \mathcal{S} et $\mathcal{E}(D_r(\mathcal{S}))$ l'estimation obtenue à l'aide de \mathcal{E} . L'estimateur \mathcal{E} converge asymptotiquement si et seulement si :

$$|\mathcal{E}(D_r(\mathcal{S})) - \mathcal{A}(\mathcal{S})| \leq \epsilon(r) \quad \text{avec} \quad \lim_{r \rightarrow \infty} \epsilon(r) = 0$$

5.4.3.1 Solutions existantes

Dans un premier temps, MULLIKIN et VERBEEK (1993) ont proposé une approche locale pour l'estimation d'aire de surfaces discrètes. L'idée est la suivante : tout d'abord, les différentes configurations locales possibles de voxels, appartenant à une surface discrète, sont classifiées en neuf catégories. À chacune de ces classes, un poids est associé avec un processus statistique de manière à estimer au mieux l'aire de morceaux de plans discrets distribués aléatoirement. Lorsque l'on analyse une surface quelconque, nous comptons, pour chaque classe, le nombre d'occurrences du motif. Finalement, l'aire est estimée par combinaison des occurrences et des poids de chaque catégorie.

Dans une approche basée sur une approximation polyédrale de la surface discrète, deux solutions naïves peuvent être proposées (voir paragraphe 2.4.2.4) : la première consiste en le calcul de l'enveloppe convexe de l'objet discret pour ensuite estimer l'aire de cette enveloppe, la seconde se base sur une utilisation de l'algorithme des Marching-Cubes. Pour la première solution basée sur une enveloppe convexe de l'objet, l'estimateur n'est bien évidemment valide que pour des objets convexes et une preuve de convergence existe dans ce cas. Nous reviendrons sur cette dernière plus tard (SLOBODA et ZATKO, 2001). En considérant l'algorithme des Marching-Cubes, l'estimation de l'aire de la surface surface consiste à sommer l'aire de chaque facette. Cet estimateur n'a pas la propriété de convergence asymptotique compte tenu du fait qu'il s'agisse d'un calcul local. Afin d'augmenter encore la précision d'un tel estimateur dans des cas pratiques, LINDBLAD et NYSTRÖM (02) proposent une pondération des différentes configurations des facettes du Marching-Cubes. Cette dernière solution est proche de l'estimateur de MULLIKIN et VERBEEK (1993) décrit ci-dessus.

KLETTE et SUN (2001) utilisent une facettisation en plan discret et se basent sur cette dernière pour estimer l'aire de la surface. L'idée est ici de mettre en place le même type d'estimateurs que \mathcal{E}_{DR-DSS} ou \mathcal{E}_{VK-DSS} dans le cas 2D. Cependant, comme nous l'avons vu dans le paragraphe 2.4.2.4, de nombreux problèmes existent pour la localisation des sommets d'une polyédrisation et la construction des faces. Ainsi, la mesure donnée par cet estimateur est très variable en fonction des stratégies choisies (propagation, distribution de germes...). De plus, le coût en complexité des algorithmes de reconnaissance de plan discret étant assez important, l'estimateur

obtenu n'est pas très efficace sur le plan algorithmique. Finalement, aucune preuve de convergence asymptotique n'est disponible sur ce type de méthode.

Sur le plan théorique, SLOBODA et ZATKO (2001) proposent l'utilisation d'une enveloppe convexe relative comme approximation polyédrale de l'objet discret. Cette structure géométrique serait équivalente au MLP dans le cas 2D. En se basant sur leur définition de l'enveloppe convexe relative, les auteurs prouvent la propriété de convergence asymptotique de l'estimateur d'aire de surface basé sur ce principe. Cependant, aucun algorithme n'est encore disponible pour calculer une telle surface polyédrale. Finalement, YU et KLETTE (2002) proposent un algorithme permettant d'approximer cette enveloppe convexe relative dans une approche par coupes : dans chaque coupe, selon un axe de la grille, un polygone MLP est calculé. Ensuite, une surface polyédrale est construite en se basant sur les polygones MLP des coupes. Cet algorithme ne construit pas exactement l'enveloppe convexe relative de SLOBODA et ZATKO, et n'a pas la propriété de convergence asymptotique.

Comme nous pouvons le voir, les approches polyédrales pour l'estimation de surface n'offrent pas de solutions pertinentes du point de vue de la convergence asymptotique. Dans le paragraphe suivant, nous détaillons un estimateur d'aire de surface, proposé initialement par LENOIR (1999), dont nous prouvons la convergence asymptotique dans le cas général.

5.4.3.2 Estimation d'aire de surface discrète par intégration de normales

Pour faire une comparaison avec le cas bidimensionnel, cette approche correspond aux estimateurs \mathcal{E}_{TAN} et \mathcal{E}_{TAN_PIX} .

Comme nous l'avons présenté précédemment, nous avons la formule classique en géométrie :

$$\mathcal{A}(\mathcal{S}) = \int_{\mathcal{S}} \vec{n}(s) ds$$

Nous pouvons proposer, de la même manière qu'en 2D, l'estimateur d'aire de surface suivant :

$$\mathcal{E}_{TAN_SURF}(D_r(\mathcal{S})) = \sum_{s \in D_r(\mathcal{S})} \vec{n}^*(s) \cdot \vec{n}_{el}(s)$$

où $\vec{n}^*(s)$ correspond à une estimation du vecteur normal centré sur l'élément de surface s de la discrétisation de \mathcal{S} , $\vec{n}_{el}(s)$ correspond au vecteur normal élémentaire en s (voir figure 5.10).

Cet estimateur d'aire de surface a été proposé initialement par LENOIR (1999). Dans ce qui suit, nous évaluons ce dernier autant sur le plan théorique qu'expérimental.

Si la surface discrète est définie en tant qu'ensemble de voxels, et si les normales sont estimées en chaque voxel, nous pouvons décrire un estimateur similaire basé sur la mesure suivante, donnée en chaque voxel v :

$$a(v) = \frac{\|\vec{n}^*(v)\|}{\max(|n_x^*(v)|, |n_y^*(v)|, |n_z^*(v)|)}$$

Et ainsi obtenir l'estimateur suivant :

$$\mathcal{E}_{TAN_SURF_VOX}(D_r(\gamma)) = \sum_{v \in D_r(\mathcal{S})} a(v)$$

Dans notre analyse théorique, nous ne distinguons pas ces deux estimateurs.

Nous étendons le travail de LENOIR (1999) avec le théorème suivant :

Théorème 5.9 *Soit \mathcal{S} une surface C^2 dans \mathbb{R}^3 à courbure bornée, l'estimateur d'aire de surface discrète basé sur l'intégration discrète des vecteurs normaux converge asymptotiquement si et seulement si l'estimation des normales converge.*

La preuve de ce théorème est donnée en annexe B (théorème B.6 et corollaire B.3).

En utilisant le théorème 5.2, nous avons :

Corollaire 5.2 *Soit \mathcal{S} une surface C^2 dans \mathbb{R}^3 à courbure bornée, l'estimateur d'aire de surface discrète basé sur l'intégration discrète des vecteurs normaux, et dont les normales sont données par l'algorithme décrit dans le paragraphe 5.3.2.3, converge asymptotiquement.*

Ainsi, nous avons un algorithme d'estimation d'aire de surface en deux étapes :

1. Estimation des normales en chaque surfel de la surface discrète
2. Sommation, pour chaque surfel, du produit scalaire entre la normale estimée et la normale élémentaire en ce surfel.

Nous obtenons donc un estimateur :

- optimal en temps puisque l'étape 1 s'effectue en $O(n)$ où n est le nombre de surfels, et puisque l'étape 2 a aussi un coût en $O(n)$ (voir paragraphe 5.3.2.3).
- qui a la propriété de convergence asymptotique pour une famille de surface réelle vaste (voir annexe B).

5.4.3.3 Analyse expérimentale

Dans cette analyse expérimentale, nous illustrons le comportement, non pas de l'estimateur basé sur le calcul optimal dont l'implémentation finale est en perspective, mais de celui basé sur le calcul de normales présenté dans le paragraphe 5.3.2.1.

Nous comparons l'estimateur d'aire de surface discrète basé sur l'intégration de normales à, d'une part, l'estimation basée sur le calcul d'enveloppe convexe dans le cas d'objets discrets convexes, et d'autre part à l'estimation basée sur l'algorithme des Marching-Cubes.

Nous considérons les objets multi-grilles suivants (voir figure 5.30) :

- des sphères discrètes ;
- des cylindres dont l'axe principal est parallèle à l'axe de la grille ;
- des demi-sphères creusées par des sphères de rayon plus petit.

Ces objets sont donc considérés sur des grilles de résolution croissante. La mesure utilisée pour évaluer ces estimateurs est l'erreur relative en pourcent.

La figure 5.31 présente les résultats sur les sphères discrètes. Nous pouvons remarquer que l'erreur des estimateurs basés sur l'enveloppe convexe et sur l'intégration des normales décroît vers 0 pour des résolutions croissantes. De plus, nous pouvons

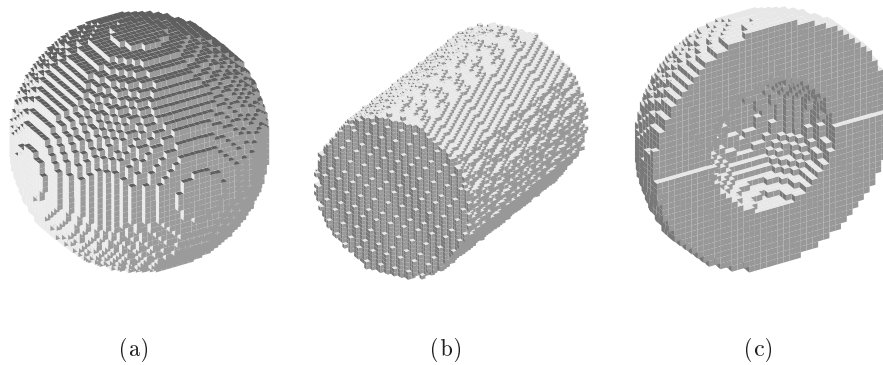


FIG. 5.30 —: *Volumes discrets multi-grilles considérés pour l'analyse expérimentale des estimateurs d'aire.*

remarquer que les résultats obtenus en intégrant les normales théoriques ou en intégrant les normales estimées par l'algorithme du paragraphe 5.3.2.1 sont quasiment confondues. L'approche basée sur l'algorithme des Marching Cubes semble avoir une erreur qui se stabilise vers 10%.

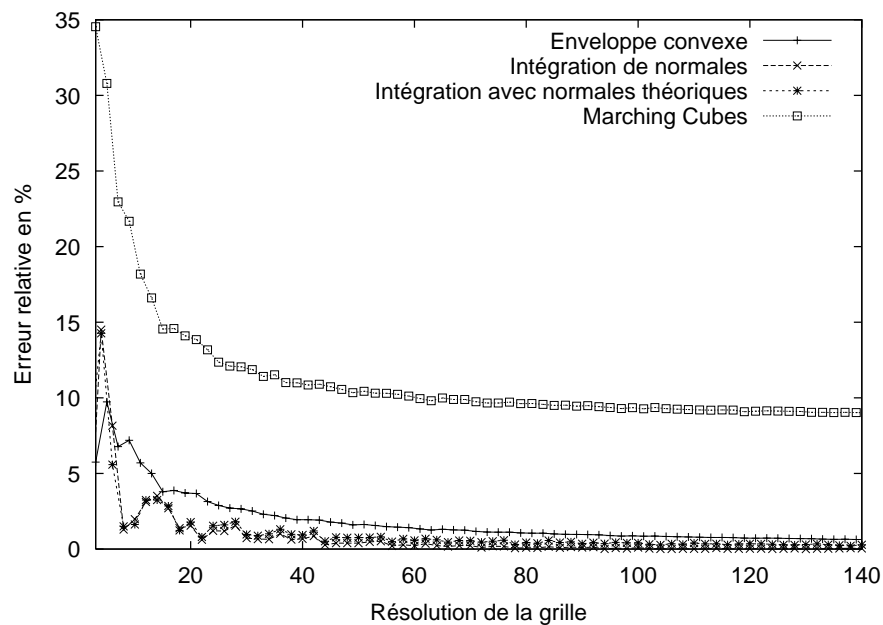


FIG. 5.31 —: *Convergence asymptotique expérimentale des différents estimateurs sur des sphères multi-grilles.*

Les figures 5.32 et 5.33 présentent les différents résultats sur les cylindres et les sphères trouées. Encore une fois, nous pouvons observer le bon comportement de notre estimateur.

Finalement, la figure 5.34 la mesure $a(v)$ associée à chaque voxel et qui correspond à l'apport du voxel v à l'aire globale de la surface. Par définition de cette mesure, celle-ci est comprise entre 1 et $\sqrt{3}$.

Ces résultats ont été présenté dans COEURJOLLY ET AL. (2002a).

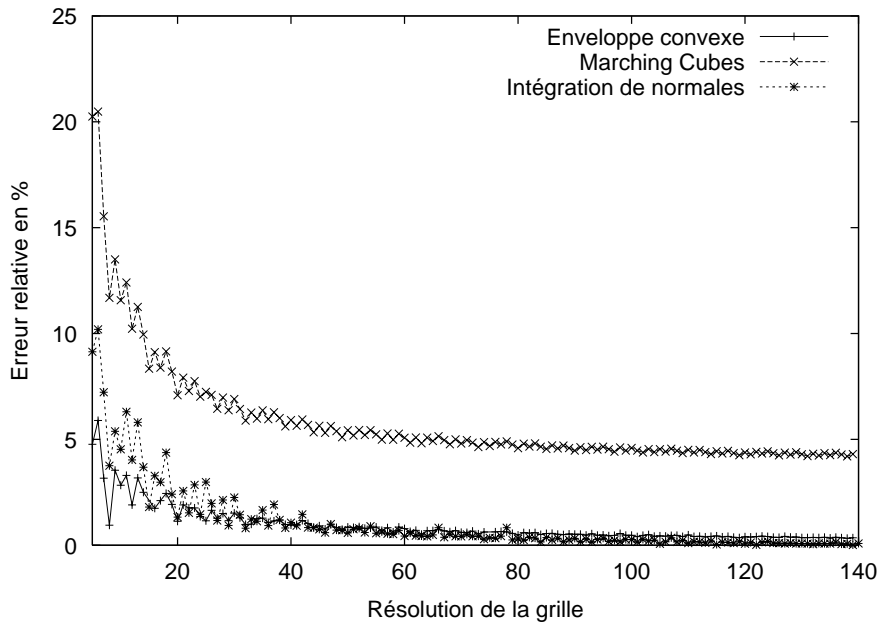


FIG. 5.32 —: *Convergence asymptotique expérimentale des différents estimateurs sur des cylindres multi-grilles.*

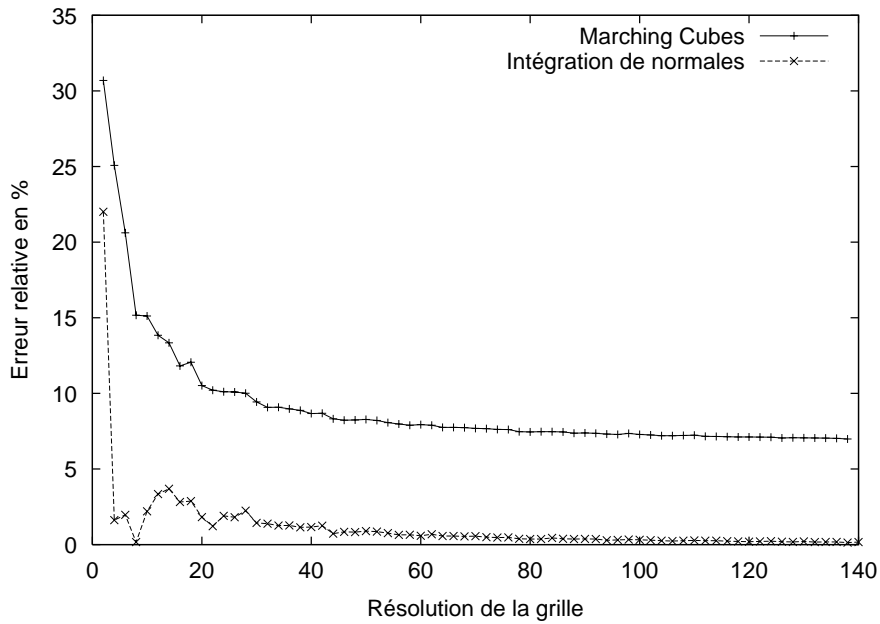


FIG. 5.33 —: *Convergence asymptotique expérimentale des différents estimateurs sur des demi-sphères trouées multi-grilles.*

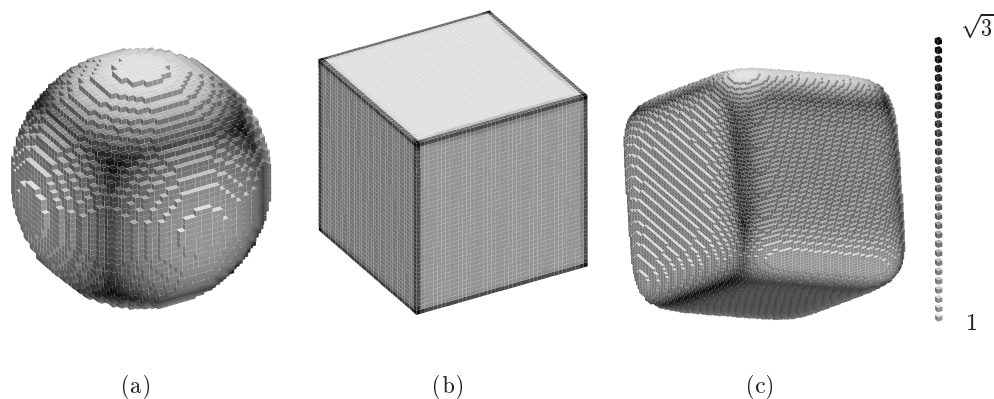


FIG. 5.34 —: Illustration de la fonction $a(v)$ représentant l'apport de chaque voxel v à l'aire globale de la surface : (a) sphère discrète, (b) sur un cube et (c) sur un cube lisse tourné.

5.4.3.4 Conclusion sur l'estimation d'aire d'une surface discrète

Sur le plan théorique, nous avons proposé un algorithme optimal en temps et qui a la propriété de convergence asymptotique pour l'estimation de l'aire d'une surface discrète.

Dans la partie expérimentale, nous avons présenté des premiers résultats d'un estimateur qui se base aussi sur l'intégration de normales discrètes mais qui ne correspond pas exactement à l'algorithme optimal.

Une perspective importante de ce travail est l'implémentation, dans une plateforme de tests, de l'algorithme optimal ainsi que d'autres algorithmes proposés dans la littérature.

5.5 Courbures d'objets discrets

5.5.1 Quelques définitions mathématiques

Dans un premier temps, nous présentons les définitions et propriétés de la courbure usuelle dans un espace euclidien. Nous verrons par la suite comment plonger ces définitions dans le modèle discret.

Supposons une courbe $x(s)$, au moins \mathcal{C}^2 et dans un espace de dimension n , paramétrée par son abscisse curviligne :

$$\begin{aligned} x : [0, 1] &\rightarrow \mathbb{R}^n \\ s &\rightarrow x(s) \end{aligned}$$

La courbure, notée $k(s)$, en un point s de x est un scalaire qui mesure la déviation de la courbe par rapport au vecteur tangent. Étant donnée une paramétrisation curviligne de la courbe, nous avons :

$$k(s) = \left| \frac{d^2 x}{ds^2} \right| \quad (5.3)$$

Ce scalaire est bien évidemment toujours positif. Cependant, pour des courbes planes, c'est-à-dire pour $x : [0, 1] \rightarrow \mathbb{R}^2$, nous pouvons signer cette valeur est ainsi

définir la *courbure algébrique* notée k_a et donnée par :

$$\frac{d\vec{t}}{ds} = k_a \vec{N}$$

avec $\vec{t}(s)$ le vecteur tangent à x en s et \vec{N} tel que (\vec{t}, \vec{N}) soit une base orthonormée.

La courbure algébrique peut donc être positive, négative ou nulle et permet de caractériser la concavité ou la convexité locale de x .

Nous avons une définition équivalente de la courbure : si nous considérons $\phi(s)$ l'angle orienté entre le vecteur tangent $\vec{t}(s)$ et une droite fixe, nous avons :

$$k_a(s) = \frac{d\phi}{ds} \quad (5.4)$$

Nous avons une troisième définition donnée par :

$$k(s) = \frac{1}{\rho(s)} \quad (5.5)$$

où $\rho(s)$ est défini comme étant le rayon du cercle qui géométriquement s'ajuste au mieux à la courbe en s . On parle alors de *cercle osculateur*.

Bien évidemment, ces trois écritures de la courbure sont totalement équivalentes. En effet, le cercle osculateur se définit à partir de la dérivée seconde de la courbe. Nous présentons ces différentes approches puisque dans le cas discret, chacune d'entre-elles va engendrer une classe d'estimateurs discrets de courbure.

Considérons maintenant une surface paramétrée $x(u, v)$ dans \mathbb{R}^3 donnée par :

$$\begin{aligned} x &: U \rightarrow \mathbb{R}^3 \\ (u, v) &\rightarrow x(u, v) \end{aligned}$$

avec U un ouvert de \mathbb{R}^2 .

Pour définir une courbure en tout point de cette surface, nous devons introduire un repère en tout point de celle-ci : soit $m = (u, v)$ un point de la surface, nous considérons le repère de FRENET est défini par la normale \vec{N} et un vecteur \vec{t} appartenant au plan tangent à la surface en m (voir figure 5.35). Dans ce repère orthonormé, nous considérons deux courbes $x_1(s)$ et $x_2(s)$ sur la surface contenues dans deux plans Π_1 et Π_2 orthogonaux et, contenant la normale en m (cf figure 5.35). Nous pouvons définir la *courbure moyenne* H en m en fonction des courbures $k_1(m)$ et $k_2(m)$ des courbes x_1 et x_2 :

$$H(m) = \frac{k_1(m) + k_2(m)}{2}$$

Si maintenant nous prenons x_1 (resp. x_2) comme étant la courbe de courbure maximale k_{max} (resp. courbure minimale k_{min}) dans toutes les directions du repère de FRENET. Nous pouvons donner la définition de la *courbure gaussienne* K de la surface au point m :

$$K(m) = k_{max}(m)k_{min}(m)$$

La connaissance de ces deux courbures nous permet de caractériser toutes les configurations locales de la surface (cf tableau 5.2 et figure 5.36).

Nous pouvons remarquer que le calcul de la courbure gaussienne est, en termes de complexité, beaucoup plus difficile à mettre en place que la courbure moyenne qui bénéficie de la propriété d'invariance quelque soit les directions du repère de FRENET.

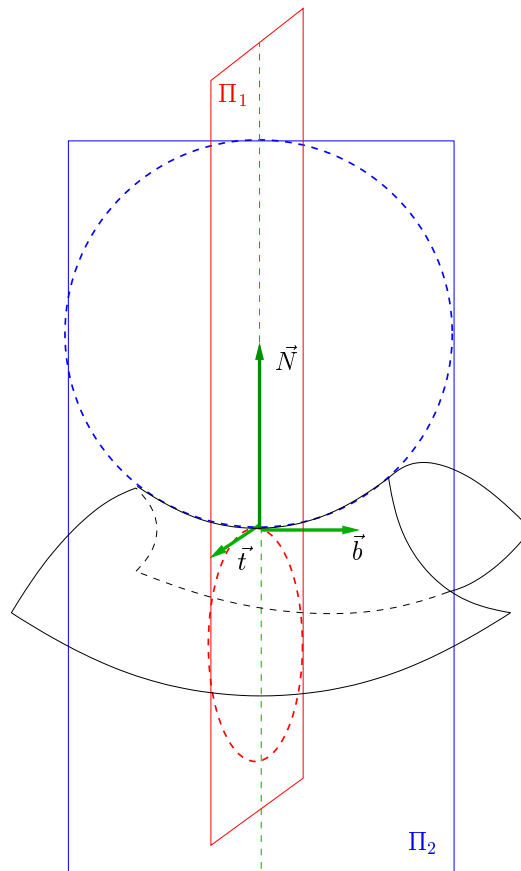


FIG. 5.35 – Exemple de plans principaux (Π_1 et Π_2) à une surface avec leur cercle principal respectif et le repère de FRENET (\vec{N} , \vec{b} , \vec{t}) ainsi défini.

	$K = 0$	$K > 0$ (elliptique)	$K < 0$ (hyperbolique)
$H = 0$	plan	indéterminé	surface minimale
$H > 0$	cylindre concave	vallée	col concave
$H < 0$	cylindre convexe	pic	col convexe

TAB. 5.2 – Caractérisation locale des surfaces en fonction des courbures moyenne et gaussienne.

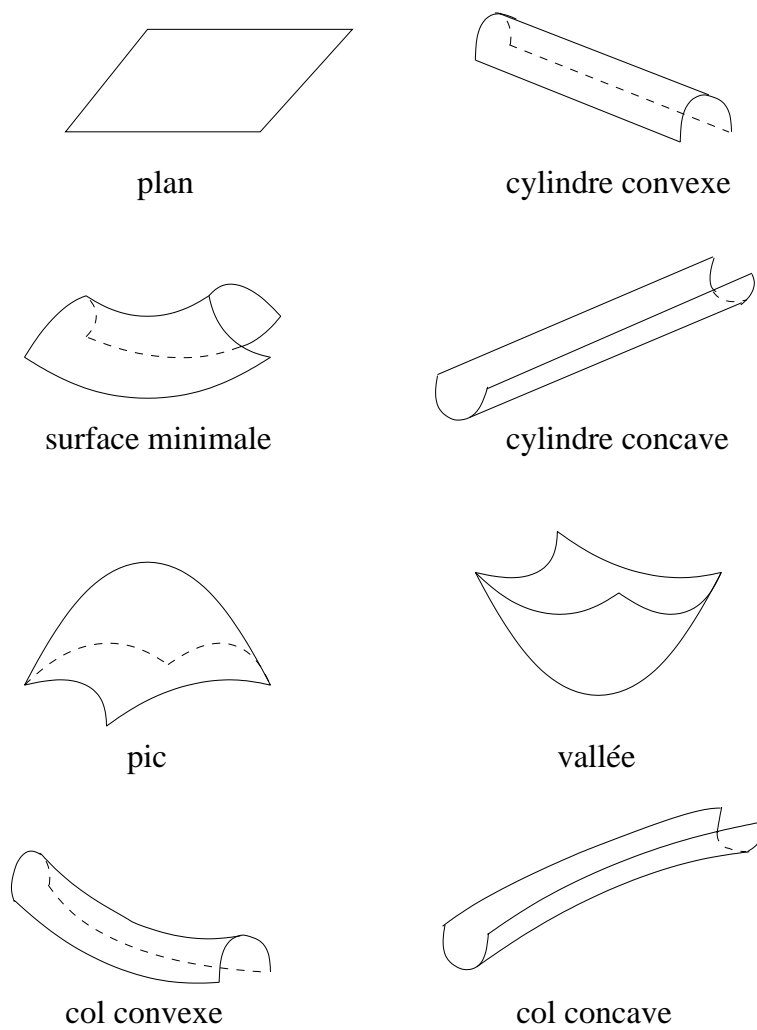


FIG. 5.36 –: *Caractérisation locale des surfaces en fonction des courbures moyennes et gaussiennes.*

5.5.2 Courbure de courbes discrètes

Nous nous intéressons dans ce paragraphe au calcul de la courbure en tout point d'une courbe discrète 2D ou 3D. Cette mesure euclidienne est une information très importante figurant dans la *signature* associée à un contour discret.

Les différents estimateurs que nous proposons correspondent essentiellement aux travaux présentés dans COEURJOLLY ET AL. (2001d) ainsi que dans COEURJOLLY ET AL. (2001c).

5.5.2.1 État de l'art

Dans cette étude bibliographique, nous reprenons l'analyse de WORRING et SMEULDERS (1993) en ajoutant les derniers travaux dans ce domaine. Dans ce qui suit, les estimateurs proposés correspondent tous à des courbures algébriques (donc signées). Le signe peut parfois être ajouté en post-traitement par une analyse de concavité ou convexité locale de la courbe en un point donné.

Ainsi, les équations (5.3), (5.4) et (5.5) ont respectivement introduit des estimateurs basés sur :

1. un calcul de dérivée seconde de la courbe discrète ;
2. un calcul de tangente puis un processus de calcul de variation d'angle des tangentes ;
3. l'estimation ou la reconnaissance du cercle osculateur.

Une des premières solutions pour l'estimation de courbure en un point p_i d'une courbe discrète se base sur la formule suivante (voir CHASSERY et MONTANVERT 1991) :

$$k^*(p_i) = \frac{\angle(p_{i-m}p_i, p_i p_{i+m})}{|p_{i-m}p_i| + |p_i p_{i+m}|}$$

Cette estimation de la courbure est appelée *courbure normalisée d'ordre m* (voir figure 5.37). Dans cette approche, le paramètre m représente un compromis entre précision et lissage du calcul.

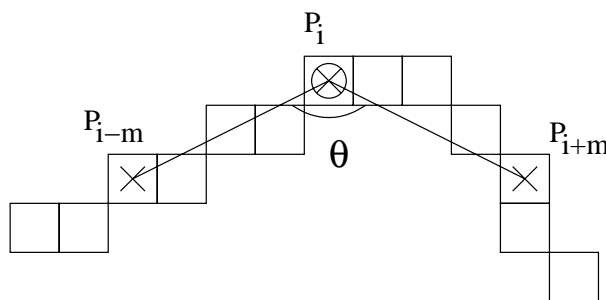


FIG. 5.37 – Exemple de calcul de courbure normalisée d'ordre m pour $m = 4$.

En se basant sur la formule (5.4), WORRING et SMEULDERS (1993) proposent l'estimateur suivant :

$$k^*(p_i) = \frac{\theta(p_i) * G'_\sigma}{1.107}$$

où :

$$\theta(p_i) = \angle(\text{LINEFITTING}(p_i, m), O_x)$$

et \mathcal{G}'_σ un filtre Gaussien dérivatif d'écart type σ .

Cette approche consiste à estimer une tangente à la courbe en p_i grâce à une technique des moindres carrés sur les points $\{p_{i-m} \dots p_{i+m}\}$ (noté LINEFITTING dans la formule). Ensuite, la fonction $\theta(p_i)$ correspond à l'angle entre cette tangente estimée au point p_i et une droite quelconque (ici l'axe O_x). Finalement, une convolution de cette fonction avec un filtre Gaussien dérivatif permet d'estimer la dérivée de θ . Le facteur de régularisation 1.107 correspond à la distance moyenne entre deux pixels de grille sur une distribution aléatoire de droites.

Par la suite, VIALARD (1996b,a) remplace l'estimation aux moindres carrés de la tangente par la tangente discrète calculée au point p_i , la formule de l'estimateur reste inchangée. D'un point de vue algorithmique, l'optimisation de FESCHET et TOUGNE (1999) permet une estimation de la courbure en tout point de la courbe en temps optimal, c'est-à-dire $O(n)$ pour une courbe de n pixels.

Dans une approche basée sur l'estimation de cercles osculateurs, WORRING et SMEULDERS (1993) proposent la formule générale suivante :

$$k^*(p_i) = \frac{1}{r^*(p_i)}$$

avec

$$(c^*, r^*) = \text{ARCFITTING}(p_i, m)$$

En d'autres termes, une estimation du centre c^* et du rayon r^* du cercle osculateur est calculée par moindres carrés (processus noté ARCFITTING) sur une fenêtre de m pixels et la valeur de la courbure est donnée par l'inverse du rayon estimé.

Finalement, WORRING et SMEULDERS (1993) proposent aussi un estimateur basé sur le calcul de la dérivée seconde de la courbe discrète. Ils montrent cependant que l'erreur engendrée par l'estimation de la dérivée d'ordre deux est assez importante.

Dans ce qui suit, nous allons nous intéresser dans un premier temps à une définition purement discrète de la courbure basée sur l'estimation ou la reconnaissance de cercle osculateur discret. En d'autres termes, de la même manière que VIALARD a résolu le LINEFITTING en discret, nous souhaitons résoudre l'ARCFITTING en discret. Ensuite, nous présentons un estimateur de courbure basé sur la notion de courbure normalisée mais avec l'introduction d'objets discrets que nous appelons *les demi-tangentes discrètes* en un point. Pour ce dernier estimateur, nous présentons une preuve de convergence asymptotique de la mesure.

5.5.2.2 Estimation du cercle osculateur discret

Nous nous intéressons à la résolution du problème ARCFITTING de WORRING et SMEULDERS. Pour cela, nous allons utiliser l'analyse faite au chapitre 3.

La première solution que nous présentons correspond à une estimation du cercle osculateur très rapide basée sur l'analyse des paraboles discrètes associées aux cercles discrets (voir paragraphe 3.3.2). Nous proposons des premières idées sur un estimateur basé sur la reconnaissance de cercle discret.

Reprenons l'analyse du paragraphe 3.3.2. Le cercle discret arithmétique de centre $(0, 0)$ et de rayon R peut-être construit par le faisceau de paraboles :

$$\begin{cases} y = \left[\sqrt{x(1+2k) - k^2 + 1/4} \right] \\ x = R - k \\ 0 \leq k \leq R \end{cases}$$

Si nous considérons une hauteur l , les cercles arithmétiques dont la première parabole a une hauteur l ont un rayon entier compris entre r_{inf} et r_{sup} donnés par :

$$r_{inf} = \lceil (l - 1/2)^2 - 1/4 \rceil \quad (5.6)$$

$$r_{sup} = \lfloor (l + 1/2)^2 - 1/4 \rfloor \quad (5.7)$$

Cet ensemble correspond au palier de paraboles de hauteur l (voir figure 5.38).

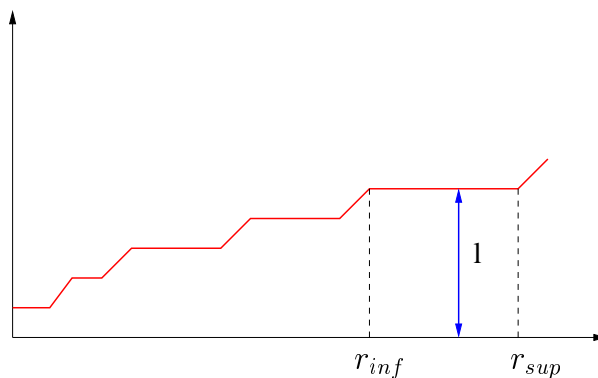


FIG. 5.38 –: *Illustration de la première parabole des cercles arithmétiques et calcul de l'ensemble des cercles de hauteur l .*

Ainsi, l'estimateur basé sur le cercle osculateur que nous proposons utilise l'analogie suivante avec le continu : considérons la tangente en un point d'un cercle de rayon r contenu dans l'anneau $r - 1/2$ et $r + 1/2$. Ce segment a une longueur constante en tout point du cercle de rayon r (voir figure 5.39). Nous partons de l'hypothèse que la discrétisation de ce segment correspond à une tangente discrète d'un cercle arithmétique de rayon r (l'anneau $r - 1/2, r + 1/2$ définissant le cercle arithmétique).

Nous émettons donc l'hypothèse que les longueurs des tangentes discrètes en tout point d'un cercle arithmétique sont constantes. Même si cette hypothèse n'est pas vraie dans le cas général, nous montrons que la variation de longueur des tangentes ne perturbe pas l'estimateur de courbure pour des rayons raisonnables.

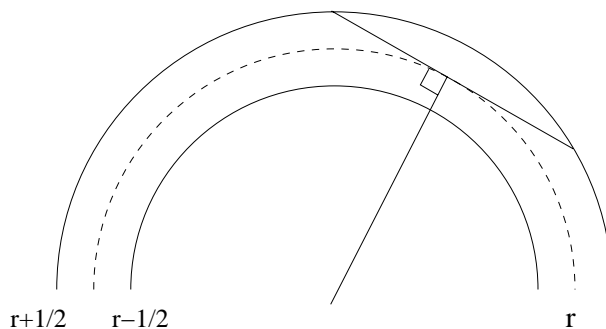


FIG. 5.39 –: *Analogie en l'anneau euclidien $(r - 1/2, r + 1/2)$, la tangente au cercle de rayon r dans cet anneau et la tangente discrète en un point d'un cercle arithmétique.*

Si nous considérons la tangente discrète au point $(r, 0)$ d'un cercle de rayon r , celle-ci correspond exactement à un segment vertical dont la longueur est le double de la hauteur de la première parabole. Ainsi, sous l'hypothèse ci-dessus, nous pouvons associer à chaque demi-longueur l de tangente discrète une estimation des cercles osculateurs discrets donnés par les formules (5.6) et (5.7).

Finalement, à partir de r_{inf} et de r_{sup} , nous estimons la courbure au point p_i comme étant l'inverse du rayon moyen :

$$k^*(p_i) = \frac{2}{r_{inf} + r_{sup}} \quad (5.8)$$

L'algorithme 25 présente l'estimateur complet.

Algorithme 25 Calcul de courbure par estimation de cercles osculateurs discrets

COURBURE_OSCULATEUR(C)

{ C est une courbe discrète}

- 1: Calcul des tangentes discrètes avec l'algorithme de FESCHET et TOUGNE (1999)
 - 2: **pour tout** pixel p_i de C **faire**
 - 3: Soit l la demi-longueur de la tangente en p_i
 - 4: Soit $r_{inf} = \lceil (l - 1/2)^2 - 1/4 \rceil$
 - 5: Soit $r_{sup} = \lfloor (l + 1/2)^2 - 1/4 \rfloor$
 - 6: $k^*(p_i) = \frac{2}{r_{inf} + r_{sup}}$
 - 7: **fin du pour**
-

Sur le plan algorithmique, nous pouvons remarquer que l'algorithme 25 s'effectue en $O(n)$ puisque les tangentes discrètes sont calculées en temps optimal (FESCHET et TOUGNE, 1999). Nous obtenons donc une estimation de la courbure très rapide et construite uniquement à partir de rayons entiers de cercles osculateurs. Notons que la valeur de la courbure calculée est, par la suite, signée en analysant la convexité ou concavité locale de la courbe discrète grâce aux paramètres des tangentes des points adjacents à p_i .

La figure 5.40 présente une première comparaison entre cet estimateur et celui proposé par VIALARD (1996a) sur un cercle de rayon 40. Nous pouvons remarquer que, sans introduction de paramètre σ toujours difficile à choisir en fonction de la donnée, nous avons une meilleure estimation de la courbure en tout point. De plus, nous notons que l'hypothèse de travail que nous avons faite, selon laquelle la longueur des tangentes discrètes en tout point d'un cercle arithmétique est constante, reste valide au vue des résultats : la variation de longueur l permet d'avoir de bon résultat sur l'estimation.

De plus, n'ayant pas de processus de lissage dans le coeur même de l'algorithme, nous avons de très bonnes localisations des points de forte courbure sur des courbes non bruitées (voir figure 5.41). Dans le cas de courbes bruitées, nous pouvons utiliser des processus de filtrage en post-traitement mais ceux-ci sont dissociés de la méthode.

Du point de vue de la convergence asymptotique, cet estimateur n'est pas du tout adapté. En effet, nous pouvons remarquer que pour des longueurs l croissantes, l'incertitude sur le rayon croît (le palier de la première parabole est de plus en plus long). Ainsi, cet estimateur permet une estimation de la courbure très rapide, assez efficace sur des objets de taille raisonnable et sans aucun paramètre.

Afin d'avoir non plus un processus d'estimation de cercle osculateur mais un processus de reconnaissance, nous pouvons envisager l'utilisation des différents algorithmes du chapitre 3. L'algorithme 26 présente une solution directe au problème du calcul de courbure par reconnaissance de cercles osculateurs. Cet algorithme repose sur la définition suivante :

Définition 5.4 (Cercle osculateur discret) *Étant donné une courbe discrète C et un point p_i de C . Le cercle osculateur discret à C en p_i est le plus grand morceau de cercle discret qu'il est possible de reconnaître, centré en p_i .*

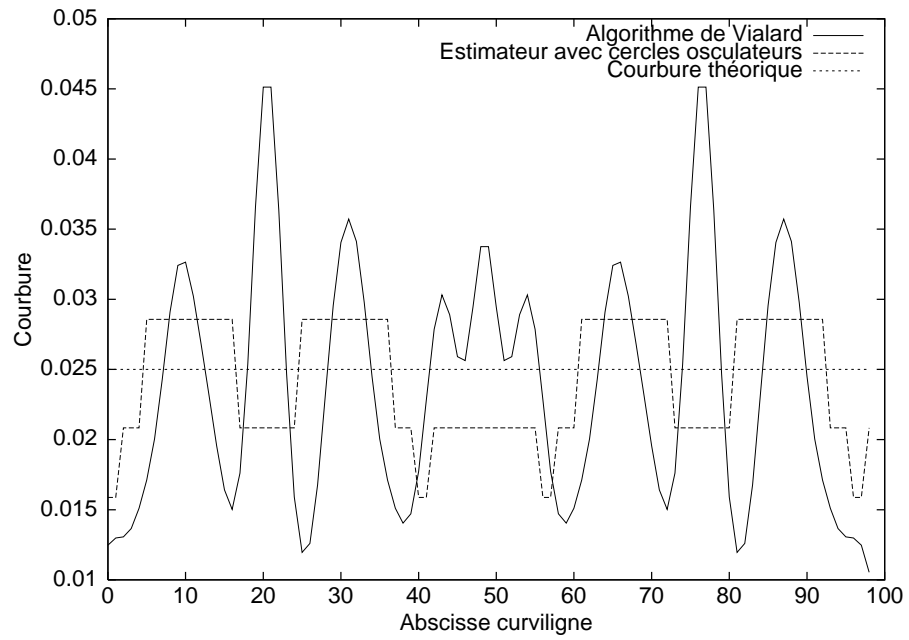


FIG. 5.40 – : Comparaison entre le graphe de courbure de VIALARD (1996a) (avec $\sigma = 1$) et l'estimation de l'algorithme 25 sur un cercle discret de rayon 40.

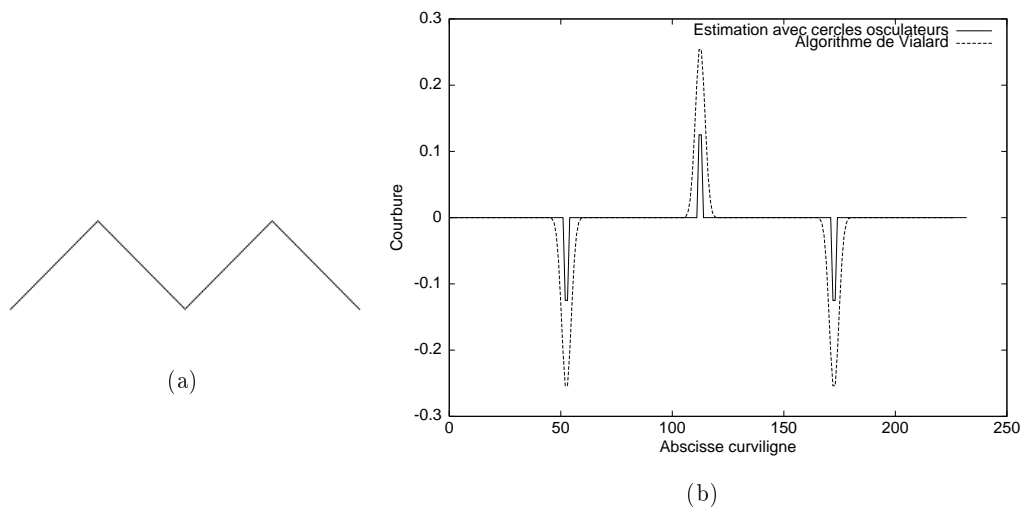


FIG. 5.41 – : Comparaison entre le graphe de courbure de VIALARD (1996a) (avec $\sigma = 1$) et l'estimation de l'algorithme 25 sur un carré déplié.

Cette définition est similaire à celle de la tangente discrète proposée par VIALLARD.

Algorithme 26 Calcul de courbure par reconnaissance de cercles osculateurs discrets

COURBURE_RECONNAISSANCE_OSCULATEUR(C)

{ C est une courbe discrète}

- 1: **pour tout** pixel p_i de C **faire**
 - 2: Reconnaître le plus grand arc de cercle centré en p_i
 - 3: Calculer, par exemple, le cercle (c^*, r^*) avec c^* barycentre du polygone acd résultat de la reconnaissance
 - 4: $k^*(p_i) = \frac{1}{r^*}$
 - 5: **fin du pour**
-

Dans les résultats de segmentation des figures 3.15 et 3.16, nous pouvons observer que la qualité de localisation du centre théorique du cercle est de l'ordre du pixel. Ainsi, ces premiers résultats laissent imaginer un très bon comportement de ce type d'estimateur de courbure.

Un inconvénient majeur de cet algorithme reste sa complexité. En effet, nous avons vu dans le chapitre 3 que la reconnaissance et la segmentation en arcs de cercles discrets d'une courbe discrète de n points ont une complexité en $O(n^{4/3} \log(n))$. Or, une implémentation directe de l'algorithme 26 nous oblige à appliquer un tel calcul en tout point de la courbe ce qui nous donnerait un coût au pire cas en $O(n^{8/3} \log(n))$.

Une perspective de cette approche serait d'analyser le comportement du polygone des solutions du cercle osculateur discret lorsque nous passons d'un point p_i au point suivant de la courbe.

5.5.2.3 Estimation de courbure avec convergence asymptotique

Dans ce paragraphe, nous présentons un estimateur de courbure qui a la propriété de convergence asymptotique. Cet estimateur repose sur une extension de la *courbure normalisée d'ordre m* (voir paragraphe 5.5.2.1 et CHASSERY et MONTANVERT 1991) avec des outils de la géométrie discrète.

L'idée principale est de construire, en un point p_i de la courbe, le plus long segment de droite discrète à droite de p_i (respectivement à gauche de p_i), le pixel extrémité de ce segment est noté P_D (respectivement P_G). Ces deux segments seront appelés respectivement **demi-tangente discrète à droite** et **demi-tangente discrète à gauche** à la courbe en p_i . Ainsi, l'estimation de la courbure au point p_i de la courbe est donnée par l'inverse du rayon du cercle passant par les points P_D , p_i et P_G .

L'estimation de la courbure se ramène donc au calcul du rayon du cercle circonscrit au triangle (P_G, p_i, P_D) . Une idée simple pour implémenter ce calcul se base sur l'intersection des médiatrices de $[P_G p_i]$ et de $[P_D p_i]$. Nous présentons cependant une formule directe permettant d'estimer très simplement le rayon R_c du cercle circonscrit à un triangle donné par les longueurs a , b et c de chacune des arêtes (HARRIS et STOCKER, 1998).

Nous calculons tout d'abord l'aire du triangle donnée par :

$$\mathcal{A} = \frac{\sqrt{(b+c)^2 - a^2} \sqrt{a^2 - (b-c)^2}}{4}$$

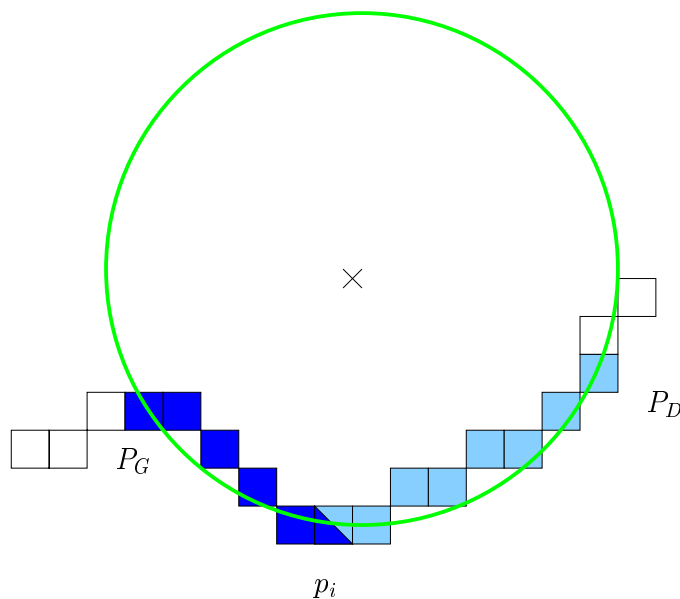


FIG. 5.42 –: Illustration de l'estimateur de courbure basé sur les demi-tangentes à gauche et à droite de p_i .

Puis nous avons la formule directe :

$$R_c = \frac{abc}{4\mathcal{A}}$$

Dans notre application, tous les cas dégénérés, qui correspondent à des demi-tangentes discrètes colinéaires par exemple, sont capturés par un simple test de nullité de \mathcal{A} (a , b et c étant non nuls par construction des demi-tangentes). Ce cas particulier nous permet de définir une courbure nulle au point p_i .

L'algorithme 27 présente l'implémentation très simple de cet estimateur de courbure. Le calcul des demi-tangentes discrètes s'effectue grâce aux différents algorithmes présentés dans le chapitre 2. Nous pouvons aussi implémenter dans cet algorithme le calcul du signe de R_c avec une simple extraction du signe du déterminant des vecteurs $\overrightarrow{p_i P_D}$ et $\overrightarrow{p_i P_G}$.

Si maintenant nous regardons le coût de cet algorithme, nous voyons qu'une implémentation directe nous donne une complexité en $O(n^2)$ pour l'étiquetage de chaque pixel par la valeur de la courbure en ce point. Cependant, nous pouvons transformer l'algorithme optimal de calcul des tangentes discrètes de FESCHET et TOUGNE (1999) pour construire en $O(n)$ toutes les demi-tangentes à gauche et à droite en tout point.

En effet, l'algorithme initial de FESCHET et TOUGNE n'impose pas la symétrie complète de la tangente dans le calcul en temps optimal : si l'ajout d'un point à gauche fait échouer la reconnaissance de segment discret, la tangente peut continuer à droite autant que possible, et réciproquement. Pour le calcul des demi-tangentes discrètes à droite, il suffit d'invalider, dès le début, le coté gauche du segment. Nous calculons ainsi en $O(n)$ toutes les demi-tangentes à droite. En appliquant un processus symétrique, nous pouvons calculer les demi-tangentes à gauche dans le même temps.

Par conséquent, l'algorithme 27 est optimal en temps puisqu'il a une complexité en $O(n)$. Notons que cette approche ne nécessite aucun paramètre pour son fonctionnement.

Algorithme 27 Calcul de courbure par demi-tangentes discrètes

COURBURE_DEMI-TANGENTES(C)
 { C est une courbe discrète}
 1: **pour tout** pixel p_i de C **faire**
 2: Calculer la demi-tangente discrète à droite $[p_i P_D]$
 3: Calculer la demi-tangente discrète à gauche $[P_G p_i]$
 4: $S = \text{signe}(\det(\overrightarrow{p_i P_D}, \overrightarrow{p_i P_G}))$ {calcul de la convexité ou concavité locale}
 5: $a = \|\overrightarrow{p_i P_D}\|$, $b = \|\overrightarrow{p_i P_G}\|$, $c = \|\overrightarrow{P_D P_G}\|$,
 6: $\mathcal{A} = \frac{\sqrt{(b+c)^2 - a^2} \sqrt{a^2 - (b-c)^2}}{4}$
 7: **si** $\mathcal{A} = 0$ **alors**
 8: $k^*(p_i) = 0$ {Cas dégénéré}
 9: **sinon**
 10: $R_c = \frac{abc}{4\mathcal{A}}$
 11: $k^*(p_i) = S \frac{1}{R_c}$
 12: **fin du si**
 13: **fin du pour**

En plus de ce résultat algorithmique, nous avons le théorème suivant :

Théorème 5.10 *Soit Γ la famille des courbes C^2 dans \mathbb{R}^2 à courbure bornée (plus quelques hypothèses techniques), l'estimateur de courbure basé sur les demi-tangentes discrètes converge asymptotiquement.*

La preuve est donnée dans l'annexe (théorème B.4).

En conclusion, nous avons présenté un estimateur de courbure basé sur la notion de demi-tangente discrète en un point d'une courbe qui est optimal en temps et qui à la propriété de convergence asymptotique.

Des analyses expérimentales préliminaires ont été effectuées pour évaluer cet estimateur mais une analyse comparative précise avec les algorithmes présentés ci-dessus est en cours de développement.

5.5.2.4 Courbures de courbes discrètes 3D

Nous présentons ici quelques solutions au problème du calcul de courbure sur des courbes discrètes tridimensionnelles. Dans la littérature, nous n'avons pas trouvé d'estimateurs existants pour un tel calcul. Ceci s'explique par le fait que peu d'applications existent dans la pratique. Récemment, ARONSSON (2002) a montré l'utilité d'une telle mesure dans un projet d'analyse de fibres de papier.

Dans ce qui suit, nous présentons des estimateurs de courbure de courbes discrètes 3D basés sur les algorithmes 25 et 27. Par la suite, nous ne parlerons que de courbure non signée, les notions de convexité ou de concavité locales n'ayant pas de sens en 3D.

Dans le cas de l'estimation du cercle osculateur basée sur la longueur de la tangente discrète (algorithme 25), nous pouvons étendre directement cette approche au cas 3D en considérant la notion de tangente discrète en un point d'une courbe discrète 26-connexe. En effet, si nous utilisons les algorithmes de reconnaissance de droites 3D vus dans le paragraphe 2.3, nous pouvons définir la notion de tangente discrète 3D comme étant le plus long segment de droite, centré au voxel considéré qu'il est possible de reconnaître. Ensuite, une fois la demi-longueur l calculée de

cette tangente, nous pouvons utiliser la formule (5.8) pour estimer la courbure en ce voxel.

D'un point de vue algorithmique, il est possible de rendre ce calcul optimal en temps. En effet, compte tenu du fait que les droites discrètes 3D se projettent sur les plans d'axe de la grille suivant des droites discrètes 2D, nous pouvons utiliser le calcul optimal des tangentes 2D pour construire la tangente 3D. Il suffit pour cela d'effectuer le calcul de tangente en temps optimal dans les trois plans de projections de la grille, puis de définir la tangente 3D à partir des résultats 2D obtenus dans les coupes.

La figure 5.43 présente un exemple de graphe de courbure calculé avec cet estimateur. Dans cet exemple, nous pouvons distinguer sur le graphe : la zone de courbure presque constante qui correspond à l'arc de cercle, la zone de forte courbure qui correspond à la jonction entre l'arc de cercle et le segment de droite et enfin un ensemble de voxels à courbure presque nulle qui correspond au segment.

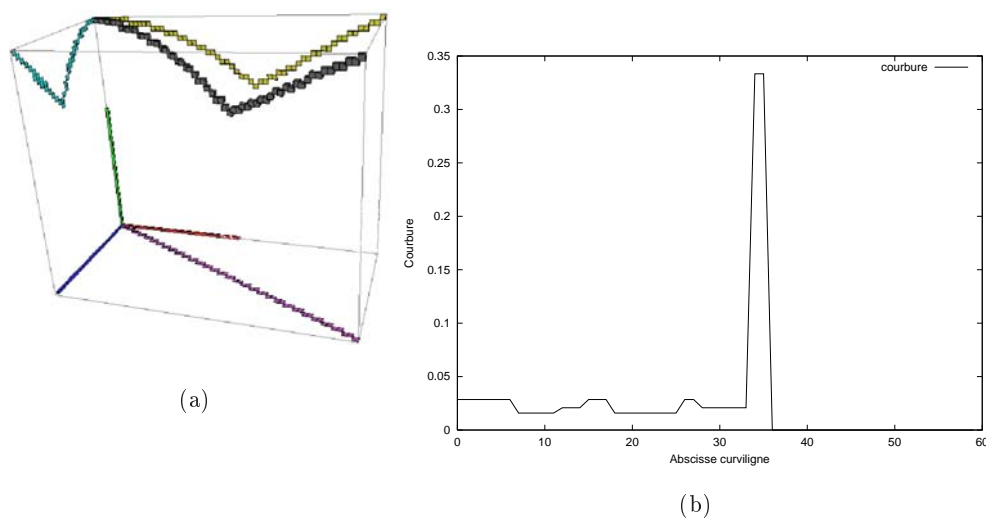


FIG. 5.43 — Exemple de calcul de courbure sur une courbe discrète 3D avec une estimation du cercle osculateur : (a) courbe discrète 26-connexe composée d'un arc de cercle et d'un segment de droite discrète 3D, (b) graphe de courbure obtenu.

Nous pouvons aussi présenter une extension de l'algorithme basée sur les demi-tangentes (algorithme 27). En effet, pour les mêmes raisons que précédemment, nous pouvons définir la notion de demi-tangente discrète 3D en un point d'une courbe discrète 26-connexe. De plus, ce qui est remarquable avec les formules de calcul du rayon R_c (voir ci-dessus), c'est que ce dernier ne nécessite que les distances entre les sommets et est défini quelque soit la dimension de l'espace dans lequel se trouve le triangle. Sur le plan algorithmique, ce calcul se fait aussi en temps optimal, c'est-à-dire en $O(n)$ pour une courbe discrète de n voxels, pour les mêmes raisons que ci-dessus.

De plus, la preuve du théorème 5.10 est valide quelle que soit la dimension de la courbe euclidienne sous-jacente (voir annexe B.) Nous avons donc le théorème suivant :

Théorème 5.11 Soit Γ la famille des courbes C^2 dans \mathbb{R}^3 à courbure bornée (plus quelques hypothèses techniques), l'estimateur de courbure basé sur les demi-tangentes discrètes converge asymptotiquement.

En conclusion, l'algorithme 28 est optimal en temps et a la propriété de convergence asymptotique.

Algorithme 28 Calcul de courbure par demi-tangentes discrètes

COURBURE_DEMI-TANGENTES(C)
 { C est une courbe discrète 3D}
 1: **pour tout** voxel v_i de C **faire**
 2: Calculer la demi-tangente discrète 3D à droite [$v_i V_D$]
 3: Calculer la demi-tangente discrète 3D à gauche [$V_G v_i$]
 4: $a = \|\overrightarrow{v_i V_D}\|$, $b = \|\overrightarrow{v_i V_G}\|$, $c = \|\overrightarrow{V_D V_G}\|$,
 5: $\mathcal{A} = \frac{\sqrt{(b+c)^2 - a^2} \sqrt{a^2 - (b-c)^2}}{4}$
 6: **si** $\mathcal{A} = 0$ **alors**
 7: $k^*(v_i) = 0$ {Cas dégénéré}
 8: **sinon**
 9: $R_c = \frac{abc}{4\mathcal{A}}$
 10: $k^*(v_i) = \frac{1}{R_c}$
 11: **fin du si**
 12: **fin du pour**

5.5.3 Courbures de surfaces discrètes

Nous nous intéressons maintenant aux différentes courbures sur surfaces discrètes 3D. Sur le plan théorique, ces mesures sont des éléments importants de la signature d'une surface discrète. D'un point de vue pratique, ces mesures ont aussi de nombreuses applications en physique des matériaux (voir annexe A).

5.5.3.1 Courbure Moyenne

Pour l'estimation de la courbure moyenne, deux approches différentes sont possibles : la première se base sur une analyse volumique de l'objet et la seconde repose sur une décomposition en coupes de la surface discrète.

Dans une première approche volumique, nous pouvons utiliser le formalisme associé aux surfaces implicites afin de proposer un estimateur volumique de courbure moyenne. En effet, si nous considérons une surface implicite donnée par l'iso-surface $f(x, y, z) = 0$, nous pouvons caractériser la valeur de la courbure moyenne en un point de cette surface à partir des dérivées partielles premières et secondes de f dans les trois directions (voir par exemple THIRION et GOURDON 1992; MONGA ET AL. 1994). Si nous reprenons l'analyse présentée dans le paragraphe 5.3.2.1 qui consiste à construire la fonction f à partir de la transformée en distance des voxels de la surface discrète, nous pouvons estimer cette courbure en calculant des dérivées partielles dans cette carte de distance.

Une autre approche volumique, moins coûteuse que la précédente, a été proposée par BULLARD ET AL. (1995). L'estimateur est assez simple et permet une mesure linéaire en la courbure moyenne. L'idée est de centrer, en tout point v de la surface discrète, une boule de rayon r . Ensuite, il faut estimer le volume V de l'intersection entre cette boule et l'objet discret en comptant le nombre de voxels de l'objet dans cette boule (la figure 5.44 illustre ce calcul dans le cas 2D). Finalement, BULLARD ET AL. (1995) proposent la formule :

$$H^*(v) = \frac{3.868}{\pi r^4} V - \frac{7.179}{3r}$$

Les différents nombres réels de cette formule ont été choisis de façon à minimiser l'erreur sur un ensemble de sphères discrètes. La figure 5.45 présente quelques résultats d'étiquetage par la courbure moyenne sur différents objets. Le niveau de gris est proportionnel à la valeur $H^*(v)$: les niveaux clairs correspondent aux parties convexes de la surface et les couleurs sombres aux zones concaves. Étant donné que les poids sont optimisés sur des sphères, nous remarquons que les centres des faces des cubes sont étiquetés "faiblement concaves" avec l'estimateur précédent. Bien évidemment, la qualité de l'approximation dépend aussi du choix du rayon de la boule r en fonction des objets à analyser.

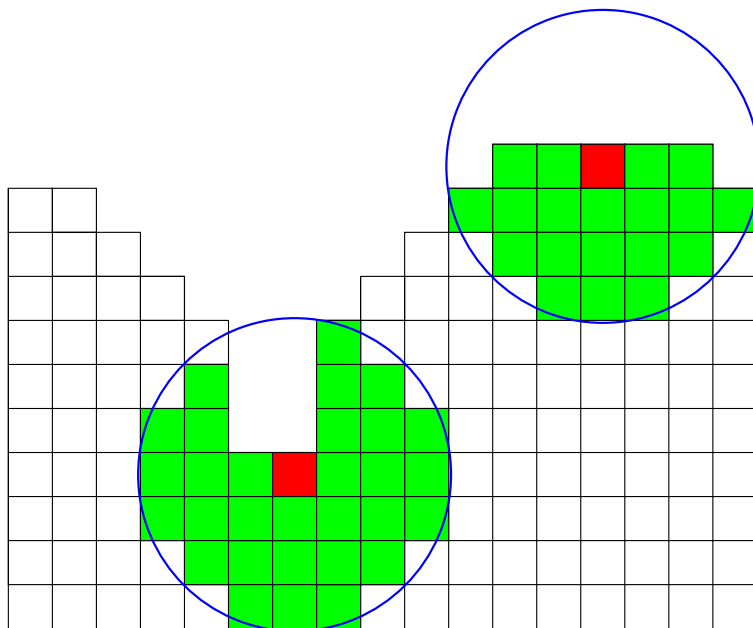


FIG. 5.44 – : *Illustration en 2D de l'estimateur de courbure moyenne de BULLARD ET AL. (1995).*

Dans une approche surfacique, LENOIR (1999) propose une décomposition de la surface discrète en coupes pour le calcul de la courbure moyenne. Ainsi, étant donné un surfel s de la surface discrète, on commence par calculer la valeur de la courbure 2D en s dans les deux coupes associées à ce surfel. Soient $k_1(s)$ et $k_2(s)$ ces deux valeurs, et $\vec{n}(s)$ la normale en s .

Rappelons que nous avons :

$$H(m) = \frac{k_1(m) + k_2(m)}{2}$$

si $k_1(m)$ et $k_2(m)$ sont calculées sur des courbes contenues dans des plans orthogonaux contenant la normale en m .

Dans cette approche par coupes, deux possibilités s'offrent à nous :

- la normale a une ou deux de ses composantes nulles, dans ce cas les deux coupes sont des plans orthogonaux contenant la normale. Nous utilisons donc l'estimateur :

$$H^*(s) = \frac{k_1(s) + k_2(s)}{2}$$

- la normale a ses trois composantes non nulles, les plans de coupe sont orthogonaux mais ne contiennent pas la normale. La formule précédente ne s'applique pas.

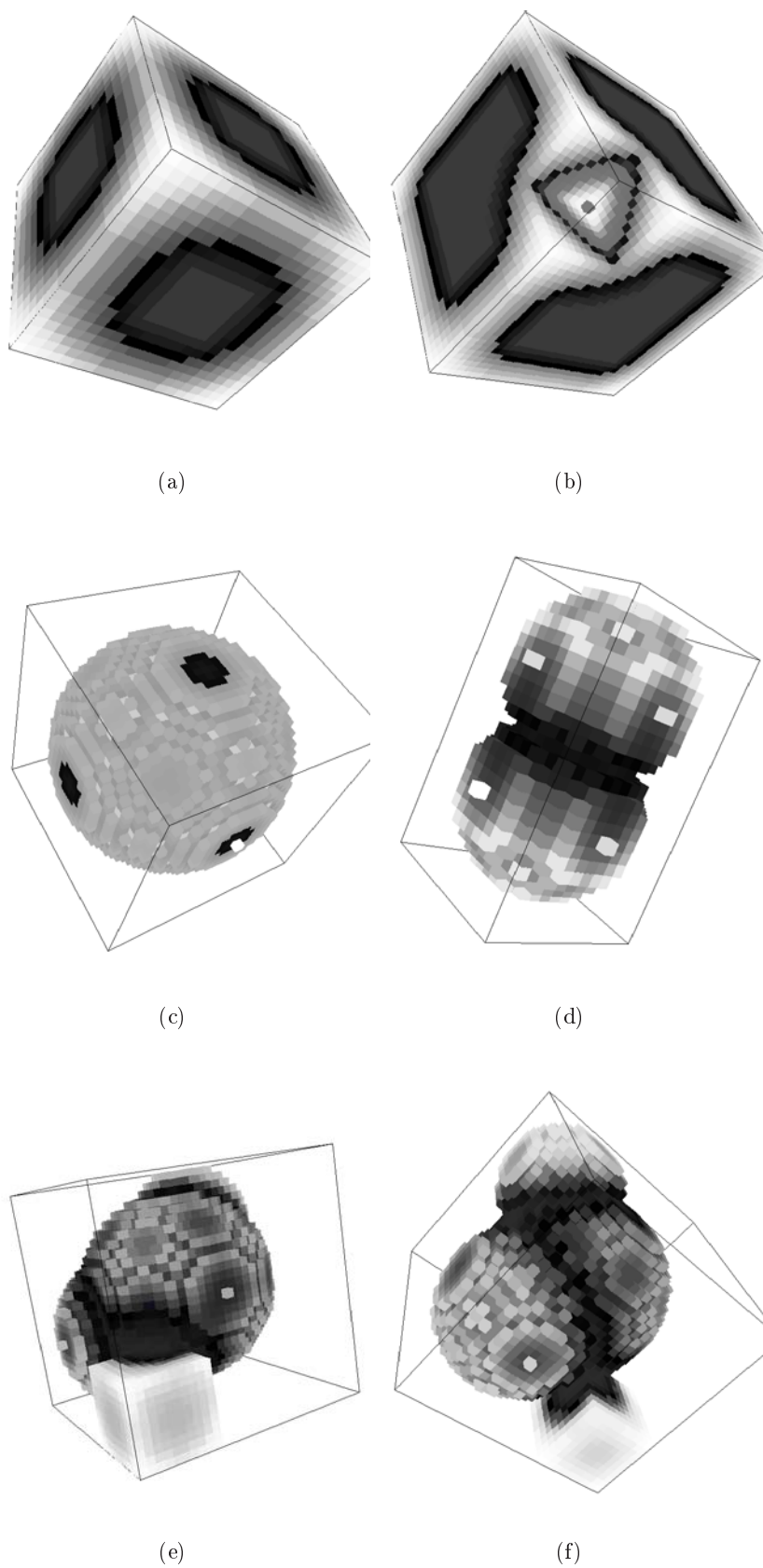


FIG. 5.45 – Exemples de calcul de courbure moyenne avec l'estimateur de BULLARD ET AL. (1995) : (a) sur un cube discret, (b) sur un cube dont un coin est chanfreiné, (c) sur une sphère discrète, (d) sur un objet composé de deux sphères connexes et (e) – (f) sur un objet composite.

Pour résoudre ce dernier point, LENOIR (1999) propose l'utilisation de la formule d'EULER : si nous connaissons les courbures k_1 , k_2 et k_3 en un point m , de trois courbes adjacentes en m , la courbure moyenne est alors une combinaison linéaire de ces courbures :

$$H(m) = w_1 k_1 + w_2 k_2 + w_3 k_3$$

avec $\{w_i\}_{1..3} \in \mathbb{R}$.

En se basant sur cette formule, LENOIR (1999) construit trois courbures, les deux premières correspondent aux courbures dans les coupes $k_1(s)$ et $k_2(s)$, la troisième correspond à une estimation basée sur une analyse locale en s .

5.5.3.2 Courbure Gaussienne

Sur le plan bibliographique, peu de solutions existent pour l'estimation de courbure Gaussienne sur des surfaces discrètes. Dans le cas de la courbure moyenne, différentes recherches ont été menées en physique des matériaux (voir par exemple BULLARD ET AL. 1995) puisque cette information a un sens en terme d'équilibre d'interface.

La courbure Gaussienne quant à elle ne semble pas avoir une signification physique directe. Cependant sur le plan de la description de forme, elle est une mesure prépondérante. Nous proposons donc un estimateur de courbure Gaussienne qui permet un calcul rapide de cette mesure.

Cet estimateur se base sur le calcul de courbure grâce à l'application de GAUSS :

Définition 5.5 (Application de GAUSS) *L'application de GAUSS, notée \mathcal{G} , associe à tout point d'une surface M orientable et au moins C^1 , son vecteur normal orienté. Cette application est illustrée par une injection entre les points de M et les points à la surface de la sphère unité.*

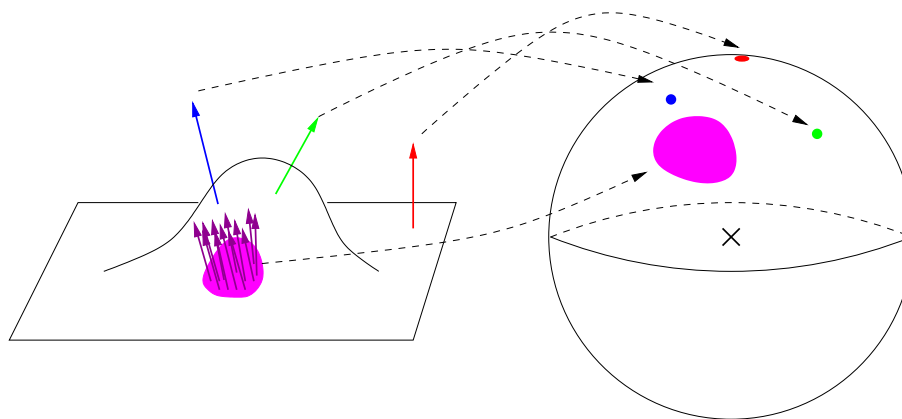


FIG. 5.46 – Illustration de l'application de GAUSS dans le cas de surfaces dans \mathbb{R}^3 .

De plus, nous avons le théorème suivant en géométrie différentielle classique :

$$K(m) = \lim_{v \rightarrow m} \frac{\mathcal{A}(\mathcal{G}(v))}{\mathcal{A}(v)}$$

où v est un ouvert sur la surface contenant m tendant vers ce dernier. $\mathcal{A}(v)$ est l'aire de ce voisinage sur la surface et $\mathcal{A}(\mathcal{G}(v))$ l'aire de l'ouvert sur la sphère unité de l'application de GAUSS \mathcal{G} de v .

Dans le cas des surfaces triangulées, cette relation entre la courbure et l'aire a permis la construction d'estimateurs très simples. Par exemple, la figure 5.47 illustre

un estimateur simple de courbure Gaussienne en un sommet d'une triangulation (voir par exemple BOIX (1995)). Cet estimateur consiste à envoyer les normales des facettes adjacentes au sommet sur la sphère unité et ensuite à faire le rapport entre la somme des aires des triangles et l'aire du polygone sphérique engendré par les normales sur la sphère unité.

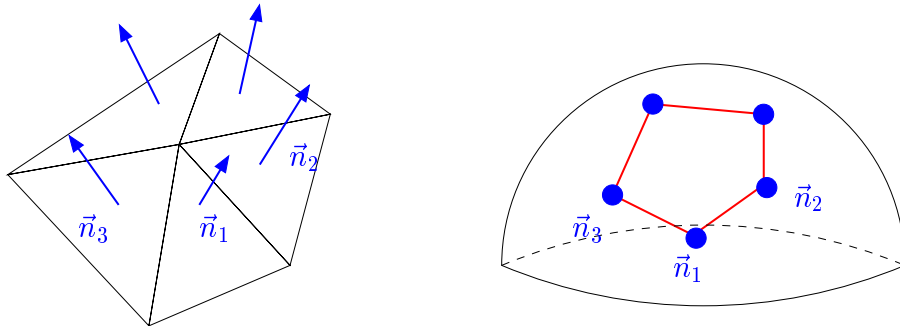


FIG. 5.47 –: Application de GAUSS et estimation de courbure Gaussienne dans le cas d'une surface triangulée.

Nous présentons brièvement cet estimateur mais une importante théorie mathématique existe pour prouver une sorte de convergence asymptotique de ce dernier (BOIX, 1995).

Dans le cas discret, une application directe de cet estimateur n'est pas réaliste. En effet, celle-ci consisterait en la projection sur la sphère unité des normales issues des surfels adjacents à un certain pointel de la surface. Étant donné que le nombre de telles configurations locales est faible, la mesure de courbure ainsi obtenue n'aurait pas un très bon comportement.

Cependant le fait d'avoir beaucoup de normales (une par surfel) pour caractériser une petite partie de la surface va nous permettre de proposer des estimateurs très efficaces. L'idée est la suivante : considérons une surface sur laquelle nous avons estimé la normale en tout surfel. Pour un surfel s de la surface, nous nous intéressons au voisinage géodésique de normales autour de s paramétré par une distance d (voir paragraphe 4.4.4). Nous considérons un processus, pour l'instant abstrait noté $\text{AIRE_SPHÈRE}(\mathcal{G}(v))$, qui nous permet d'estimer l'aire associé à un ensemble fini de points sur la sphère unité donné par l'application de GAUSS des normales dans le voisinage v . Nous définirons plus formellement ce processus par la suite. Enfin, nous utilisons l'estimateur de surface basé sur l'intégration de normales du paragraphe 5.4.3 pour estimer l'aire $\text{AIRE}(v)$ de la surface associée au voisinage v . Nous proposons donc l'estimateur :

$$K^*(s) = \frac{\text{AIRE_SPHÈRE}(\mathcal{G}(v))}{\text{AIRE}(v)}$$

Par rapport aux approches basées sur des surfaces triangulées, nous considérons un ensemble de normales dans un voisinage plus grand qu'une simple adjacence avec le surfel s . Le paramètre d , fixant le rayon géodésique de ce voisinage, permet un contrôle de l'estimateur.

Le problème maintenant est donc d'estimer l'aire associée au nuage de points $\mathcal{G}(v)$ sur la sphère unité. La figure 5.48 nous illustre le problème avec les deux solutions que nous avons apporté.

Nous définissons l'aire de ce nuage de points comme étant l'aire du plus petit polygone sphérique convexe contenant ces points. Avant de poursuivre, le choix de

se baser sur la notion d'enveloppe convexe n'est pas anodine. En effet, au sens de la géométrie différentielle, l'application de GAUSS sur un ouvert convexe reste convexe sur la sphère unité. Or, notre définition de disque géodésique surfacique semble proche d'une notion de convexité sur la surface euclidienne sous-jacente (une preuve formelle de ce point est en cours de rédaction).

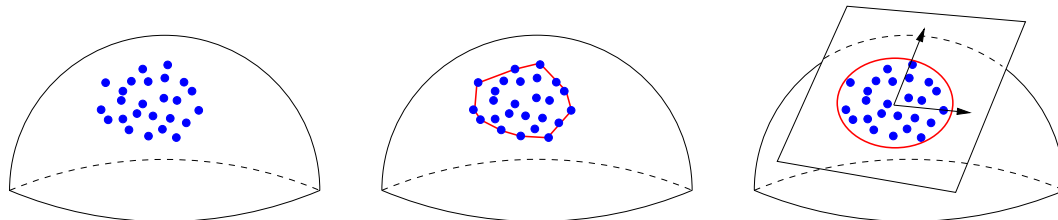


FIG. 5.48 –: *Illustration des différentes solutions pour l'estimation de l'aire d'un nuage de points sur une sphère : (de gauche à droite) les données du problème, solution par calcul d'enveloppe convexe sphérique, solution par calcul d'une ellipse de meilleure approximation.*

La première solution est très géométrique et se base sur la notion d'enveloppe convexe sphérique. Avant cela, rappelons quelques principes de la géométrie sur la surface d'une sphère. On appelle *grand cercle* l'intersection de celle-ci avec un plan contenant son centre. Le rayon d'un grand cercle correspond au rayon de la sphère. Un *triangle sphérique* (ou *triangle d'EULER*) est tel que ses trois arêtes correspondent à des arcs de grands cercles distincts. Enfin, un *polygone sphérique* correspond à une généralisation des triangles sphériques : les arêtes correspondent à des arcs de grands cercles. Un angle d'un triangle ou d'un polygone sphérique en un sommet correspond à l'angle entre les plans générant les grands cercles associés aux arêtes sphériques du sommet.

Finalement, pour ce qui nous intéresse, l'aire d'un polygone sphérique P de n sommets et dont la somme des angles est Θ , est donnée par la formule très simple :

$$\text{AIRE_SPHÈRE}(P) = (\Theta - (n - 2)\pi) R^2$$

où R est le rayon de la sphère (ici $R = 1$).

En se basant sur ces notions, nous avons proposé l'algorithme 29. L'idée est d'utiliser l'algorithme de calcul d'enveloppe convexe 2D de GRAHAM (voir par exemple CORMEN ET AL. 1990; DE BERG ET AL. 2000) pour l'étendre au cas sphérique.

Ainsi, basé sur la notion d'angle présentée ci-dessus, nous pouvons construire un tri polaire sphérique étant donné un centre et surtout, nous pouvons construire un prédicat CONVEXE qui étant donnés trois sommets décide si l'angle sphérique formé est convexe ou concave. Ce prédicat nous permet d'implémenter l'algorithme de GRAHAM. Cet algorithme a pour hypothèse que tous les sommets sont dans la même demi-sphère. Dans le cas contraire nous pouvons avoir des ambiguïtés dans la définition du polygone sphérique.

Cet algorithme est assez simple sur le plan théorique mais son utilisation pour l'estimation de courbure rend son implémentation assez difficile. En effet, dans le cas où les normales sont issues d'un morceau de sphère discrète par exemple, les points $\mathcal{G}(v)$ sont assez bien répartis et l'algorithme peut fonctionner sans difficulté (voir figure 5.49-(a)). Par contre, dans le cas d'une surface plutôt plane ou plutôt cylindrique, la courbure Gaussienne associée est alors nulle et donc les points $\mathcal{G}(v)$

Algorithme 29 Aire d'un nuage de points sur la sphère par enveloppe convexe sphérique

AIRE_SPHERE_ENVELOPPE($\mathcal{G}(v)$)

{ $\mathcal{G}(v)$ est le nuage de points sur la sphère unité}

- 1: Choisir un point p_0 de $\mathcal{G}(v)$ dont les coordonnées sphériques (θ, ϕ) sont minimales (ordre lexicographique)
 - 2: Trier $\mathcal{G}(v)$ dans un ordre polaire sphérique de centre p_0
 - 3: Calculer l'enveloppe convexe sphérique par l'algorithme de GRAHAM
 - 4: Calculer la somme des angles Θ du polygone résultant
 - 5: **retourner** $\Theta - (n - 2)\pi$
-

sont, soit tous regroupés en paquets, soit étirés dans une direction (voir figure 5.49-(b)). Dans ce cas, de nombreux problèmes numériques surviennent dans le calcul du tri polaire et dans le calcul de l'enveloppe convexe sphérique.

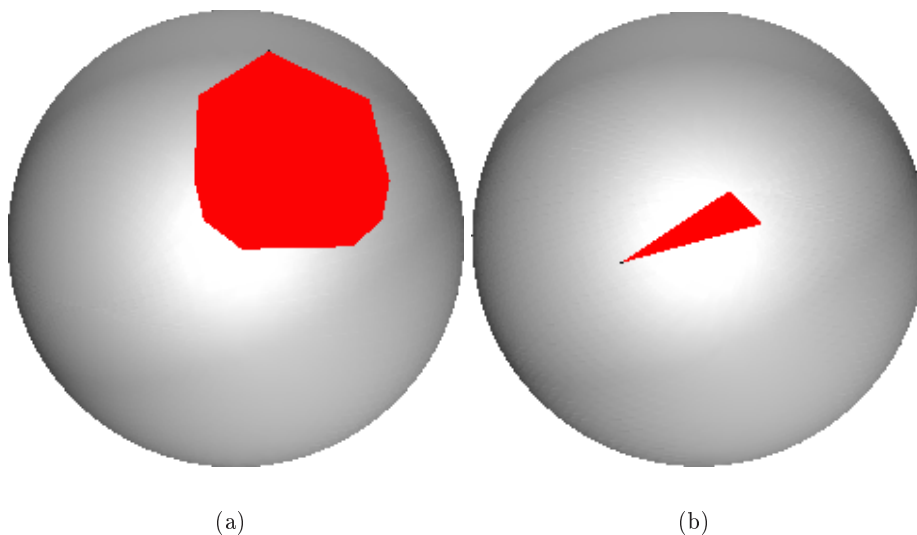


FIG. 5.49 – Exemples de calcul d'enveloppes convexes sphériques : (a) représentation sur la sphère unité du polygone obtenu sur un voisinage v provenant d'une sphère discrète et (b) provenant d'un morceau de cylindre.

Pour résoudre ces problèmes d'instabilité numérique, nous proposons une autre approche basée sur l'ellipse de meilleure approximation. Rappelons le principe de cette approximation dans le cas 2D. Étant donné un ensemble de N points $\{(x_i, y_i)\}_{i=1..N}$ du plan, nous calculons la matrice d'inertie :

$$M = \begin{bmatrix} I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{bmatrix}$$

où :

$$I_{xx} = \frac{1}{N} \sum_{k=1}^N (x_k - \mu_x)^2$$

$$I_{yy} = \frac{1}{N} \sum_{k=1}^N (y_k - \mu_y)^2$$

$$I_{xy} = \frac{1}{N} \sum_{k=1}^N (x_k - \mu_x)(y_k - \mu_y)$$

$$\mu_x = \frac{1}{N} \sum_{k=1}^N x_k$$

$$\mu_y = \frac{1}{N} \sum_{k=1}^N y_k$$

Ensuite, l'ellipse de meilleure approximation est définie comme étant l'ellipse de centre (μ_x, μ_y) et dont les longueurs des axes sont données par :

$$a_i = \sqrt{\lambda_i}$$

où les λ_i (pour $i = 1, 2$) correspondent aux valeurs propres de la matrice M (voir figure 5.50). Les directions des axes de l'ellipse sont données par les vecteurs propres de M . Enfin, l'aire de cette ellipse est donnée par :

$$\mathcal{A} = \pi \sqrt{\lambda_1 \lambda_2}$$

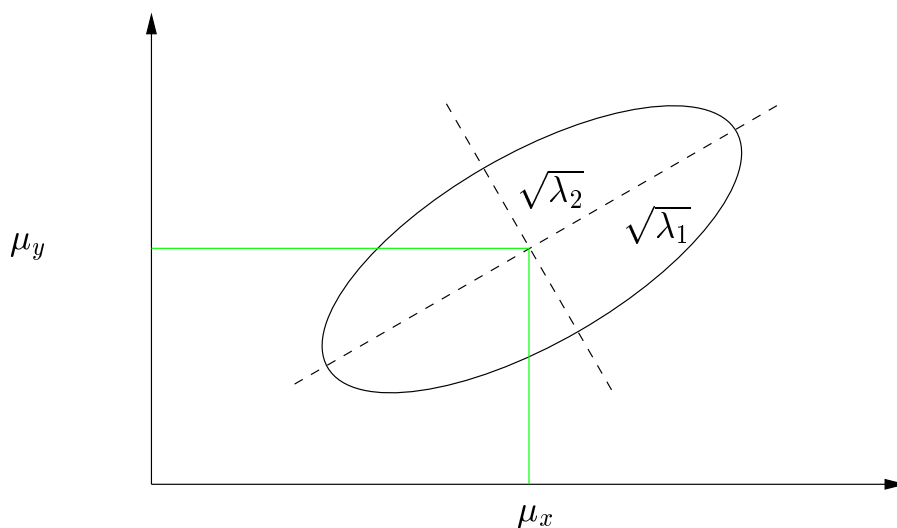


FIG. 5.50 — Ellipse de meilleure approximation dans le plan. Les directions des axes principaux sont données par les vecteurs propres de la matrice d'inertie.

Dans notre cas, l'idée est de résoudre statistiquement notre problème AIRE_SPHERE avec cette notion d'ellipse de meilleure approximation. Cependant, il n'est aisé de calculer des moments de points sur la sphère unité. Nous utilisons pour cela une approximation qui se base sur une projection du nuage de points sur un plan *représentatif*.

Ce plan correspond au plan tangent à la surface de la sphère au point qui est le barycentre du nuage de points. Nous faisons donc une approximation en trois étapes : dans un premier temps, on calcule le barycentre du champ de normales $\mathcal{G}(v)$, ensuite, nous effectuons une rotation de ces normales pour que le barycentre coïncide avec le point $(\pi, \pi/2)$ dans l'espace (ϕ, θ) . Finalement, nous utilisons une projection perspective, dont le centre est le centre de la sphère, sur le plan d'équation $x = -1$ afin d'estimer l'ellipse de meilleure approximation des projections de $\mathcal{G}(v)$. Encore une fois, ce calcul n'est valide que si les points $\mathcal{G}(v)$ sont dans la même demi-sphère. Les deux premières étapes de recentrage des normales nous permettent de simplifier le processus de projection (voir algorithme 30).

Nous obtenons ainsi une estimation de l'aire, notée \mathcal{A}_{proj} , associée au nuage de points $\mathcal{G}(v)$. Ce processus est une approximation d'autant plus juste que les points sont *regroupés* autour du barycentre. Il nous permet une estimation très rapide et avec très peu d'instabilités numériques de la courbure gaussienne.

Nous obtenons donc l'algorithme 30. La figure 5.51 présente quelques résultats de cet estimateurs sur des volumes discrets. Remarquons que dans le cas du cylindre, nous avons effectivement une valeur de la courbure Gaussienne proche de 0 partout sauf sur les bords des bases circulaires (voir tableau 5.2).

Algorithme 30 Aire d'un nuage de points sur la sphère par calcul d'une ellipse de meilleure approximation

AIRE_SPHERE_ELLIPSE($\mathcal{G}(v)$)

{ $\mathcal{G}(v)$ est le nuage de points sur la sphère unité}

- 1: Soit μ le barycentre des normales $\mathcal{G}(v)$
- 2: Appliquer une rotation aux normales pour faire coïncider μ avec $(\pi, \pi/2)$ en coordonnées sphériques
- 3: Projection des normales (x, y, z) en coordonnées sphériques $(\phi, \theta, 1)$ sur le plan $x = -1$ avec la formule :

$$X = \tan(\phi - \pi)$$

$$Y = \tan(\theta - \frac{\pi}{2})$$

- 4: Calculer l'ellipse de meilleure approximation (λ_1, λ_2) sur les points $(X_i, Y_i)_{i=1 \dots |\mathcal{G}(v)|}$
 - 5: **retourner** $\mathcal{A}_{proj}(\mathcal{G}(v)) = \pi\sqrt{\lambda_1\lambda_2}$
-

Dans les estimateurs que nous avons montrés, le signe de la courbure Gaussienne est obtenu en post-traitement. Pour cela, il suffit de considérer une courbe fermée orientée autour du surfel s considéré (la courbure Gaussienne en s doit être non-nulle) et de comparer son sens avec le sens de la courbe obtenue sur la sphère unité par application de GAUSS (voir figure 5.52).

5.6 Conclusion

Dans ce chapitre, nous avons présenté une analyse algorithmique et théorique d'estimations de mesures euclidiennes (longueur, aire, courbure...) sur des objets discrets divers (courbes 2D, 3D, surfaces discrètes...). À chaque fois que cela a été possible, nous nous sommes intéressés à des notions de convergence asymptotique des estimateurs étudiés. Pour résumer brièvement, nous avons montré la convergence asymptotique des estimateurs de longueur de courbes 2D et 3D, d'un estimateur de

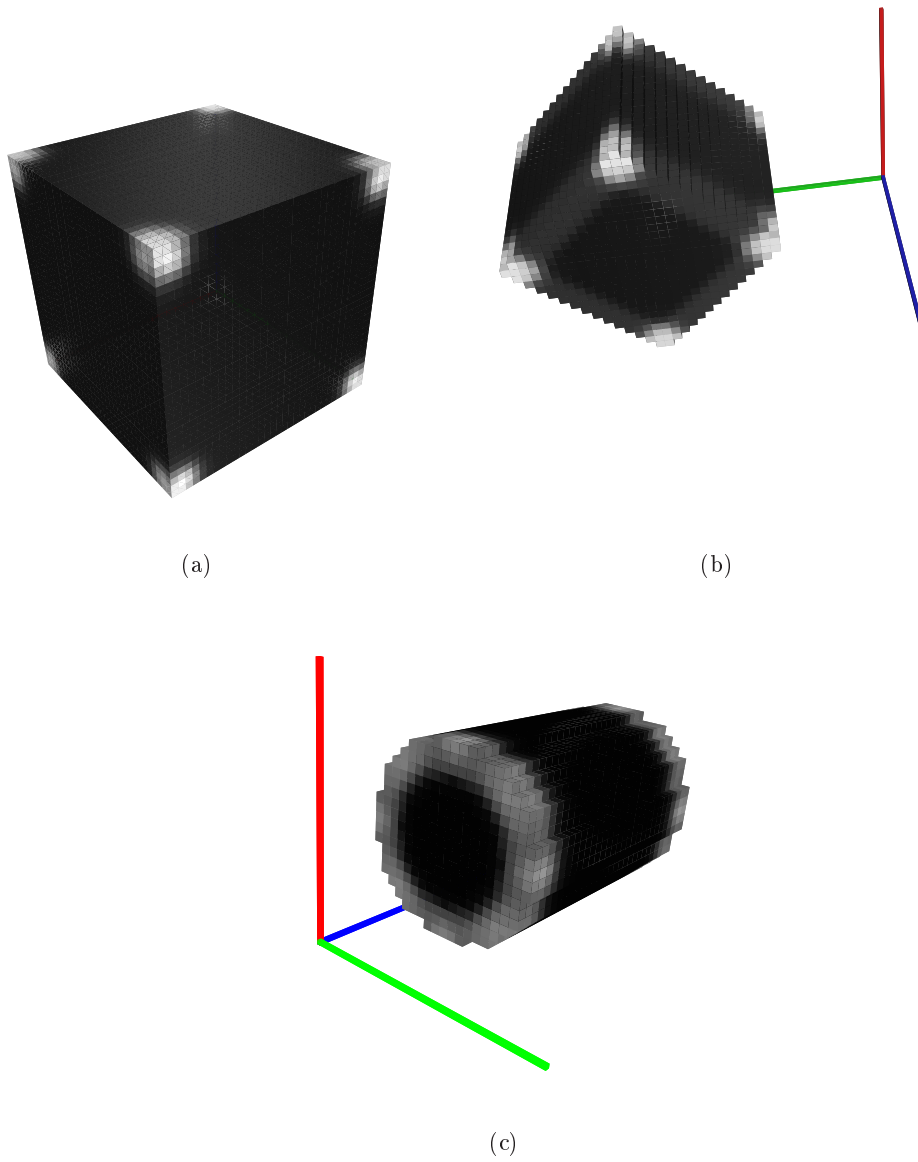


FIG. 5.51 —: Exemples de calcul de courbure Gaussienne : (a) sur un cube dans les axes, (b) sur un cube tourné et (c) sur un cylindre. Les voxels noirs correspondent à des valeurs de courbures nulles ou quasi-nulles.

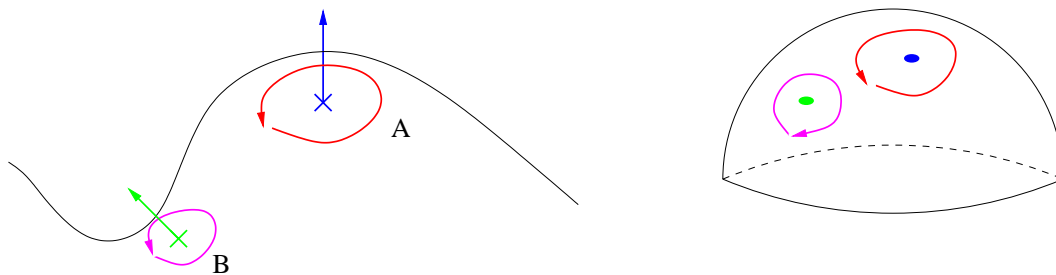


FIG. 5.52 –: *Illustration du calcul du signe de la courbure Gaussienne : la courbe A et sa transformée par GAUSS sont dans le même sens, le signe est donc positif. Il y a changement de sens pour la courbe B, le signe sera négatif.*

normales de courbes 2D et de surfaces discrètes, d'un estimateur de courbure de courbes 2D et 3D et enfin d'un estimateur d'aire de surfaces discrètes.

Parmi toutes ces mesures, l'analyse expérimentale de l'estimation de longueur d'une courbe discrète 2D a bénéficié d'une plate-forme de tests conséquente (8 estimateurs implémentés et 5 formes multi-grilles). Une des perspectives majeures de ce chapitre repose donc la création de plates-formes similaires pour l'estimation de courbure et d'aire de surfaces.

Conclusions et perspectives

Durant cette thèse, différents problèmes de la géométrie discrète ont été abordés. Ainsi, plutôt que de présenter une conclusion et des perspectives générales sur ce travail, nous préférons détailler notre contribution par thème abordé.

Objets discrets

Résumé

Dans cette partie, nous avons tout d'abord présenté les différentes définitions des droites, plans et cercles discrets de la littérature.

Dans le cas des droites discrètes, nous nous sommes surtout intéressés à une analyse bibliographique avec une comparaison des différentes définitions et des algorithmes permettant la reconnaissance. Nous avons aussi présenté les différents outils permettant de segmenter une courbe discrète 2D ou 3D en morceaux de droites discrètes. Concernant les plans discrets, après une analyse bibliographique similaire, nous avons montré un résultat original sur la structure la pré-image associée à un morceau de plan discret. En conclusion de ces travaux sur les droites et plans discrets, nous avons proposé l'utilisation de la statistique pour la reconnaissance de ces objets. Pour cela, nous avons tout d'abord illustré les liens entre les droites et les plans discrets, et des lois classiques en statistique (loi de BERNOULLI, loi empirique...). Nous avons ensuite présenté quelques pistes d'utilisation de ces outils statistiques pour l'estimation de plans tangents discrets.

Enfin, nous nous sommes intéressés aux cercles discrets et nous avons présenté un algorithme original de reconnaissance de cercles discrets basé sur une analyse arithmétique du problème. Cet algorithme utilise des procédés très simples de la géométrie algorithmique et permet une segmentation incrémentale d'une courbe discrète en arcs de cercles.

Perspectives

Concernant les plans discrets, à la suite de notre théorème de structure de la pré-image d'un morceau de plan vérifiant certaines hypothèses, nous avons conjecturé que ce résultat est vrai en relâchant les hypothèses sur le morceau de plan considéré. La preuve de cette conjecture est donc une perspective importante à ce travail. De plus, le point de vue statistique que nous avons présenté pour caractériser des droites ou plans discrets nous semble très prometteur dans de nombreuses applications.

En ce qui concerne les cercles discrets, nous avons présenté quelques pistes pour l'utilisation de l'algorithme de reconnaissance pour calculer le cercle osculateur discret (plus long arc de cercle qu'il est possible de reconnaître centré sur le pixel

considéré) en un point de la courbe discrète. Un problème intéressant consiste en une analyse incrémentale de ce calcul de cercle osculateur en tout point d'une courbe discrète.

Transformée en distance et squelette de formes

Résumé

Nous avons tout d'abord décrit un algorithme de calcul de la transformée en distance euclidienne sans erreur. Cet algorithme est optimal en temps quelle que soit la dimension de l'image. Il utilise, en fait, une structure géométrique appelée diagramme de Voronoï discret. Nous avons proposé l'utilisation directe de ce diagramme pour le calcul du squelette d'une forme. Des premières analyses ont été faites dans le cas de formes en dimension 2. Elles nous ont permis de proposer un algorithme d'extraction de squelette linéaire en le nombre de pixels de la forme. Celui-ci se base sur le calcul du diagramme et sur un processus de filtrage des sommets et des arêtes de ce dernier.

Perspectives

Une première perspective à ce travail consiste en une comparaison de la méthode d'extraction de squelette basée sur le diagramme de Voronoï discret en dimension 2 avec les approches construisant l'axe médian sur la transformée en distance.

L'algorithme d'extraction du diagramme de Voronoï discret étant optimal en temps quelle que soit la dimension, il est normal de s'intéresser à des constructions de squelettes, basés sur celui-ci, en dimension quelconque.

Visibilité et chemin géodésique discrets

Résumé

Nous avons proposé une définition de la visibilité, entre deux pixels dans un domaine discret non-convexe, basée sur la notion de droites discrètes. Nous avons présenté un algorithme efficace permettant de trouver tous les points visibles d'un pixel donné dans un tel domaine. Nous avons, par la suite, utilisé ce calcul de visibilité pour la construction de chemins géodésiques discrets, ce qui permet de définir une métrique géodésique dans des domaines discrets.

Nous avons ensuite discuté d'une généralisation de ces définitions aux objets tridimensionnels et avons donc proposé un premier algorithme de calcul de chemins géodésiques dans des domaines 3D non-convexes et sur des surfaces discrètes.

Perspectives

Du point de vue algorithmique, une perspective consiste en une réduction de la complexité du calcul de visibilité dans des domaines 3D. Nous avons présenté quelques pistes vers un algorithme efficace mais des recherches plus approfondies sont nécessaires.

Enfin, que ce soit en dimension 2 ou 3, il nous semble très intéressant de comparer ces calculs de fronts de propagation discrets aux algorithmes classiques se basant sur une résolution numérique d'un système d'équations aux dérivées partielles.

Estimateurs de mesures euclidiennes

Résumé

Nous nous sommes intéressés aux algorithmes permettant d'estimer des mesures euclidiennes (longueur, aire, courbure...) sur des contours 2D ou 3D d'objets discrets. Ces différents algorithmes ont été analysés, à chaque fois que cela a été possible, dans un cadre pratique et théorique basé sur la notion de convergence asymptotique. Nous présentons dans le tableau 5.3 les différents estimateurs proposés dans le chapitre 5. Pour chacun de ces estimateurs, nous indiquons leur complexité et si la mesure converge asymptotiquement ou pas (n représente la taille des objets considérés et d un paramètre fixe de l'algorithme).

	Estimateur	Complexité	Convergence	paragraphe
Courbes discrètes 2D	Longueur	$O(n)$	oui	5.4.1
	Normales/Tangentes	$O(n)$	oui	5.3.1
	Courbure	$O(n)$	oui	5.5.2.3
Courbes discrètes 3D	Longueur	$O(n)$	oui	5.4.2
	Tangentes	$O(n)$	oui	5.3.1
	Courbure	$O(n)$	oui	5.5.2.4
Surfaces discrètes	Aire	$O(n)$	oui	5.4.3.2
	Normales	$O(n)$	oui	5.3.2.3
	Courbure Moyenne	$O(dn)$	non	5.5.3.1
	Courbure Gaussienne	$O(dn)$	non	5.5.3.2

TAB. 5.3 –: *Estimateurs discrets de mesures euclidiennes.*

Perspectives

Lors de l'analyse du chapitre 5, seuls les estimateurs de longueur de courbes discrètes 2D ont bénéficié d'une plate-forme de tests suffisamment stable et complète pour qu'elle soit diffusée à la communauté. Une perspective importante est de construire une telle plate-forme pour toutes les autres mesures. Un travail préalable consiste à mettre en œuvre, pour la mesure étudiée, un jeu de tests significatifs et multi-grilles (pour une évaluation asymptotique).

D'un point de vue théorique, seules les estimations des courbures surfaciques ne possèdent pas encore d'algorithme ayant la propriété de convergence asymptotique. La poursuite de l'analyse de ces estimateurs est donc aussi une perspective à courte échéance.

Applications

Nous avons présenté, dans l'annexe A, différentes applications dans lesquelles nous avons utilisé nos algorithmes. Pour l'application "Neige", la poursuite de cette collaboration avec les glaciologues du Centre d'étude de la Neige de Météo-France nous semble particulièrement intéressante. En effet, il est assez rare, en géométrie discrète, d'avoir des données binaires autres que celles provenant de segmentations d'images médicales. De plus, les estimateurs proposés dans le chapitre 5 trouvent des utilisations concrètes.

Enfin, l'application des estimateurs aux modèles déformables discrets nous semble très prometteuse.

Le mot de la fin

Ces différentes recherches ont souvent nécessité de mettre les mains dans le *campouis* informatique ou encore de faire l'ermite avec un papier et un crayon. Cependant, ce sont toujours des contacts humains avec d'autres personnes (chercheurs ou non) qui ont initié, ou fait avancer, ces recherches. Mes derniers mots vont à ces personnes.

Troisième partie

Annexes

Contenu des annexes

A Applications	231
A.1 Introduction	232
A.2 Extraction de primitives pour la classification de profils de stèles funéraires	232
A.3 Analyse microscopique d'échantillon de neige	236
A.4 Modèle déformable discret	239
A.5 Conclusion	240
B Preuves de convergence asymptotique	241
B.1 Introduction	242
B.2 Courbes en dimension d : approche par <i>tube</i>	242
B.2.1 Estimation de longueur	242
B.2.2 Estimation de normales	246
B.2.3 Estimation de la courbure	247
B.3 Surfaces : aire et normales	247
B.3.1 Estimation de normales surfaciques	248
B.3.2 Estimation d'aire de surfaces discrètes	248
C Programmation linéaire : algorithme de PREPARATA et SHAMOS	253
D Index des auteurs cités	257

Annexe A

Applications

Sommaire

A.1	Introduction	232
A.2	Extraction de primitives pour la classification de profils de stèles funéraires	232
A.3	Analyse microscopique d'échantillon de neige	236
A.4	Modèle déformable discret	239
A.5	Conclusion	240

A.1 Introduction

Dans cette annexe, nous présentons succinctement les différentes applications dans lesquelles nous avons utilisé les différents outils présentés dans les paragraphes précédents. Plus précisément, nous nous intéressons, dans un premier temps, à un processus de *vectorisation* de courbes 2D en utilisant la notion de points caractéristiques dans le graphe de courbure. Ensuite, nous présentons une application utilisant les estimateurs d'aire et de courbure sur des surfaces discrètes. Enfin nous voyons comment les estimateurs des estimateurs du chapitre 5 peuvent être utilisés dans un processus de segmentation par un modèle déformable discret.

A.2 Extraction de primitives pour la classification de profils de stèles funéraires

Ce travail s'inscrit dans le cadre du projet "Stèles" qui est une collaboration entre le laboratoire ERIC et la Maison de l'Orient Méditerranéen de l'université Lumière Lyon 2. Ce projet a pour but d'aider les archéologues à construire une classification d'un ensemble de profils de stèles funéraires.

Des images de profils ont été recueillies par les archéologues sur le terrain et obtenues par l'utilisation de peignes à dents rétractiles (ou conformètres) qui, en suivant le bord d'une stèle, épousent le bord de celle-ci. Ensuite, ce contour peut être reproduit et numérisé (voir figure A.1). Outre ces profils, nous possédons des textes écrits résumant pour chacun d'eux certaines de leurs caractéristiques géographiques ou historiques telles que le lieu, la date, le type de population.

L'étude classique des profils par les archéologues s'effectue par l'extraction de moulures et de moulurations (ensemble de moulures) caractéristiques des styles. L'objectif de ce projet est donc de proposer une extraction automatique de primitives géométriques de ces contours 2D qui seront ensuite utilisées dans un processus plus général d'extraction de connaissance.

Ce projet a démarré par une élaboration d'un ensemble de pré-traitements permettant de passer des images de la figure A.1 à des courbes discrètes strictement 8-connexes.

Ensuite, nous nous sommes attachés à extraire des points caractéristiques (points de forte courbure) de ces courbes discrètes en utilisant les techniques proposées par TEH et CHIN (1989) : étant donné le graphe de courbure de la courbe discrète, les points caractéristiques sont des extrema locaux de ce graphe dans un voisinage adaptatif. Nous avons pour cela utilisé le calcul de courbure issu de l'algorithme 25 (voir figure A.2). Par la suite, nous utilisons des processus de reconnaissance de formes pour permettre d'associer, à chaque morceau de courbe entre deux points dominants, les différentes formes suivantes :

- segment de droite avec comme information sa longueur et sa pente ;
- arc de cercle avec comme information son rayon, son centre et son orientation.

Le profil de stèle est représenté par un vecteur dans lequel ne sont codés que les points caractéristiques et les formes associées entre ces points (voir figure A.3-(c)). L'évaluation grossière des formes entre deux points caractéristiques nous permet d'avoir de bons résultats dans un processus de création d'une stèle synthétique à partir d'un vecteur (voir figure A.3).

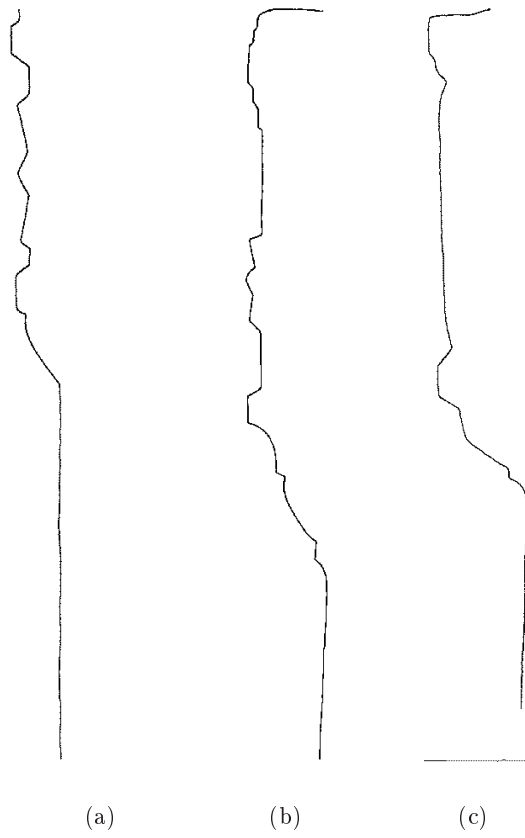
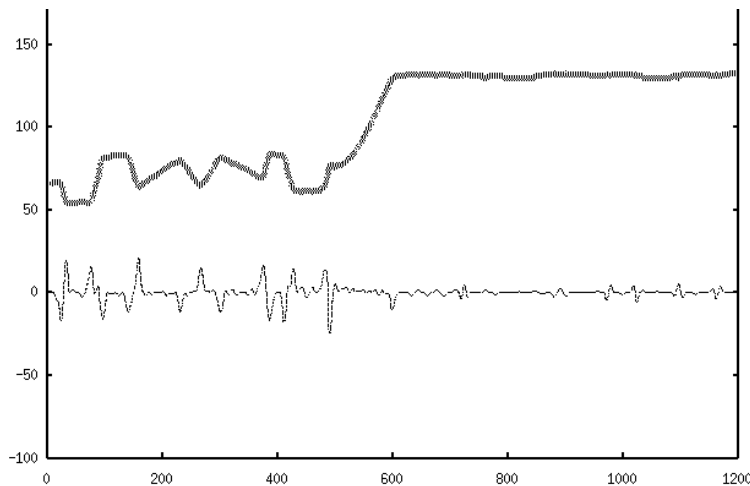


FIG. A.1 —: Exemples de profils de stèles funéraires sans pré-traitement.

FIG. A.2 –: *Exemple de graphe des courbures*

Pour avoir la possibilité d'utiliser des outils de *text mining* en plus des algorithmes classiques d'apprentissage automatique supervisé ou non supervisé, il est nécessaire de coder la description précédente sous forme de texte. Pour cela, nous construisons un alphabet dans lequel des lettres vont représenter les segments de droite selon leur orientation (lettres s_0 à s_8) et d'autres vont représenter les différentes orientations des arcs de cercles (lettres c_0 et c_1) ; la figure A.4 illustre ce codage. Il est clair que cette transformation supprime de nombreuses informations géométriques sur la stèle mais le mot obtenu est significatif de l'allure du profil.

La démarche partant d'une courbe bruitée et aboutissant à un mot est suffisamment générale pour trouver de nombreuses applications dans d'autres domaines. Une validation en apprentissage de ce codage sur les profils de stèles a débuté lors du stage de DEA de FATMA BAKLOUTI (2002) mais mérite d'être poursuivie.

Par ailleurs, dans COEURJOLLY ET AL. (2001b), le processus de reconnaissance reposait sur des approximations de segments et de cercles *épais*, et n'utilisaient pas les algorithmes décrits dans la première partie de ce manuscrit. Une perspective importante est donc d'intégrer des objets purement discrets dans ce processus.

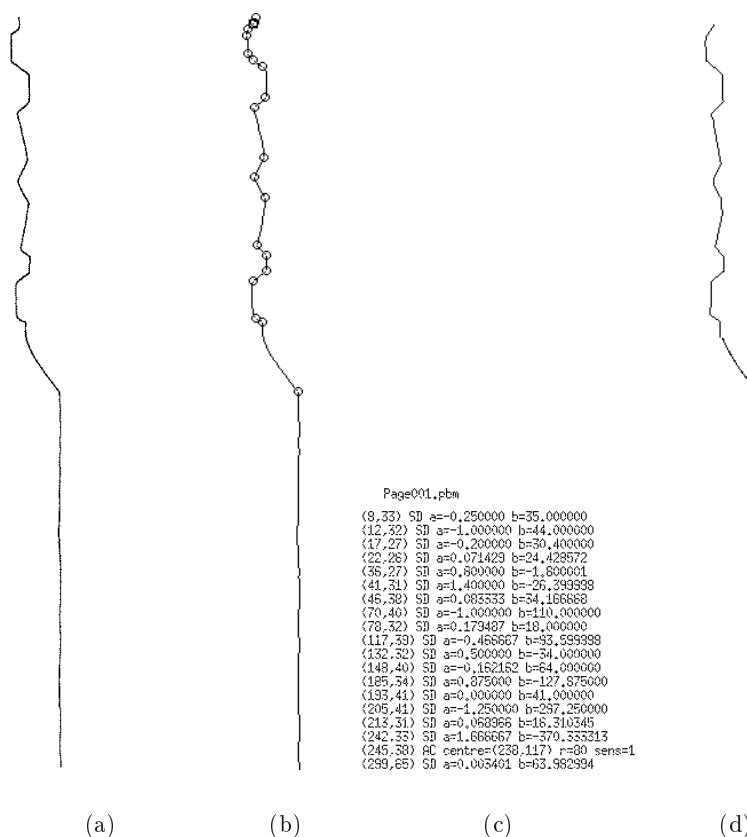
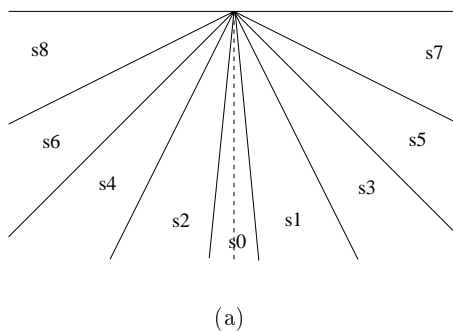


FIG. A.3 –: Vectorisation et reconstruction de profils de stèles funéraires : (a) profil de stèle sans pré-traitement, (b) après les pré-traitements et avec localisation des points caractéristiques, (c) vecteur descriptif du profil (SD pour segment de droite et AC pour arc de cercle) et (d) reconstruction à partir du code généré.



s9s5s0s6s0s3s2s3s4s9s6s0s5s0s6c1s0
 s8c1s9s3s0s3s0s4s2s0s4s0s4s0s5s10s3s4s9s4s0s5s0c2s6c1s0s4s2s0
 s0s5s0s2c1s1s3s0s4c1s1s4s1s6s0

(b)

FIG. A.4 –: Codage d'une courbe discrète après segmentation par les points caractéristiques : (a) codage utilisé pour les segments de droite et (b) exemples de mots obtenus.

A.3 Analyse microscopique d'échantillon de neige

Ce travail correspond à une collaboration avec le Centre d'Études de la Neige (CEN) de Météo-France à Grenoble. Le contexte général de ce projet est l'analyse des microstructures d'échantillons de neige afin de mieux comprendre et de modéliser le comportement du manteau neigeux. Les analyses classiques reposent sur des observations au microscope de l'échantillon de neige mais elles ne permettent pas d'observer les agencements des grains et d'étudier la répartition de l'air et de l'eau dans l'échantillon.

Pour cela, le CEN a développé un procédé permettant de maintenir en l'état un échantillon de neige fraîche jusqu'à un appareil de tomographie X afin d'acquérir une image volumique de cet échantillon (voir BRZOSKA ET AL. 1999a; FLIN 2000; COLÉOU ET AL. 2001; BRZOSKA 2001). Une fois les images acquises à l'Installation Européenne de Rayonnement Synchrotron (ESRF) à Grenoble, des traitements d'images sont effectués afin d'obtenir des volumes binaires séparant la neige de l'air. Les images de la figure A.5 présentent quelques uns de ces échantillons de neige (volumes de 300^3 voxels).

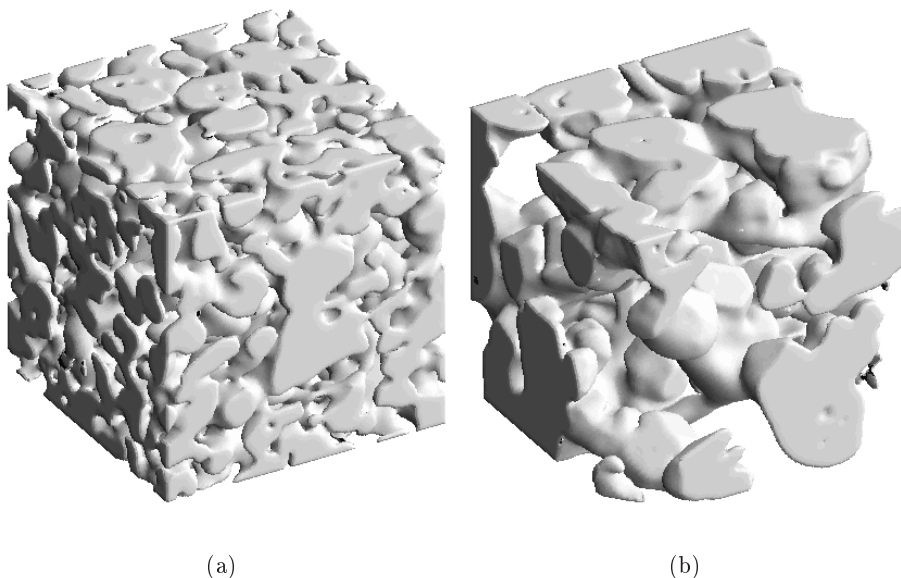


FIG. A.5 — Exemples de volumes binaires d'échantillons de neige : (a) grains fins provenant de l'Alpe d'Huez et (b) croûte de regel provenant du Col de Porte. Les volumes sont visualisés avec un rendu intégrant les normales calculées par l'algorithme présenté dans le paragraphe 5.3.2.1 (Météo-France/ESRF).

Sur ces objets discrets 3D, les mesures proposées dans le chapitre 5 ont des implications physiques et thermodynamiques importantes. Plus précisément, les estimateurs d'aire de surface discrète permettent d'approcher la notion de *surface spécifique*. De plus, à la différence des glaces profondes et mis à part les cas de forts gradients thermiques, les métamorphoses de la neige sont essentiellement contrôlées par des phénomènes d'interfaces gouvernés par la courbure moyenne (voir figure A.6).

Ainsi, un calcul efficace de courbure moyenne en tout point de tels volumes discrets permet :

- de caractériser le volume en question par son histogramme de courbure. En effet, en l'absence de gradient de température dans le manteau neigeux, les

grains grossissent et s'arrondissent lentement. Par contre, en présence d'un gradient de température, les cristaux croissent suivant leurs axes cristallographiques privilégiés et forment des facettes. Les différences entre les volumes analysés apparaissent clairement dans de tels histogrammes (COLÉOU ET AL., 2001).

- de simuler certaines métamorphoses de neige (FLIN ET AL., 2002). En effet, les zones convexes ont tendance à se sublimer tandis que la vapeur se condense dans les zones concaves.

Par la dimension des volumes manipulés, cette application nécessite des algorithmes d'estimation très robustes. Un premier résultat de cette collaboration a été la publication de travaux sur l'estimation d'aire de surfaces COEURJOLLY ET AL. (2002a). Une première étude sur l'application d'un estimateur de courbure moyenne aux échantillons de neige a permis de se rendre compte de l'intérêt d'une telle mesure (voir BRZOSKA ET AL. 1999b et figure A.6). Cependant, elle a aussi montré les différents problèmes de stabilité et d'efficacité sur des volumes importants. Des études sont en cours pour rendre un tel estimateur plus robuste et plus efficace.

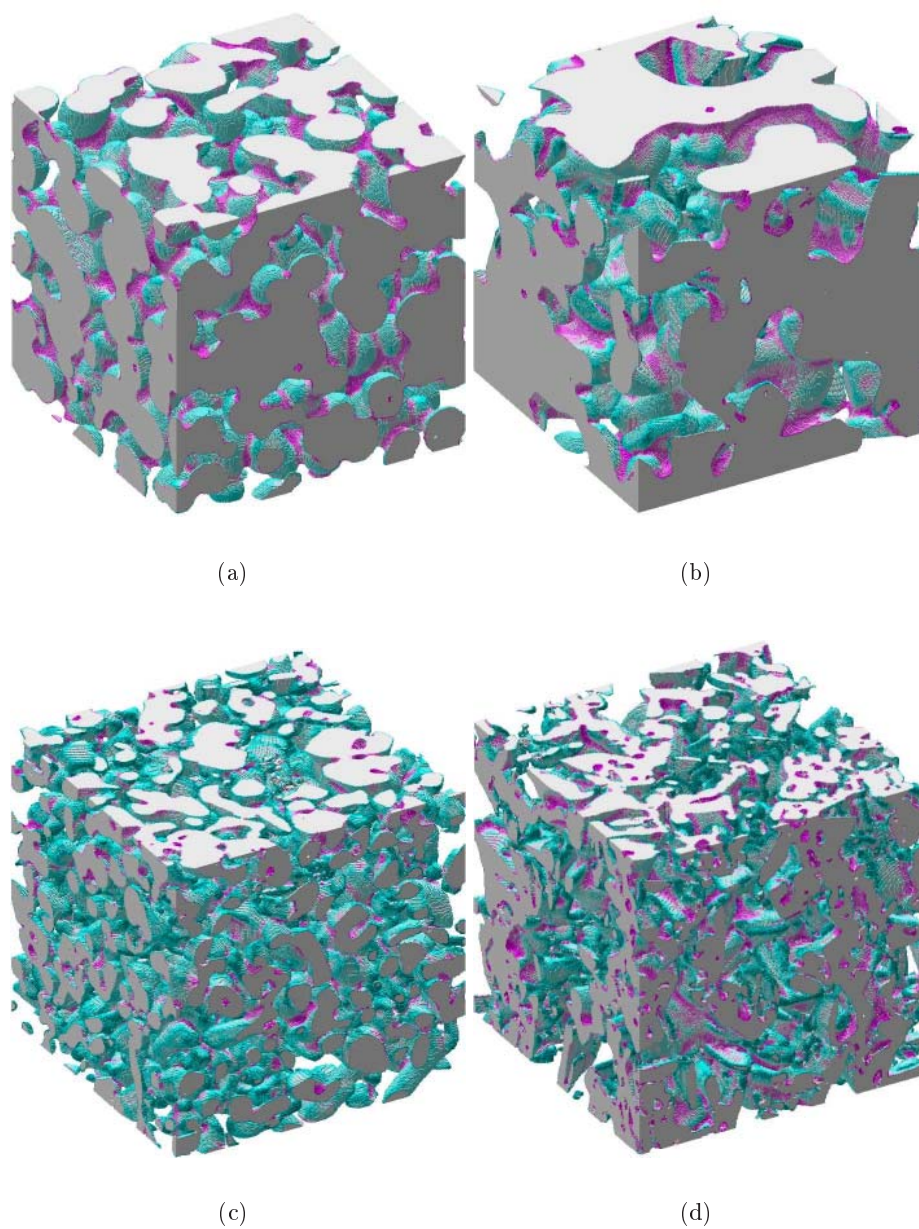


FIG. A.6 —: Exemples de coloriage des voxels à la surface d'échantillons de neige par une estimation de la courbure moyenne proposée par BRZOSKA ET AL. (1999b) (Météo-France/ESRF).

A.4 Modèle déformable discret

Ce projet est une collaboration entre le Laboratoire Bordelais de Recherche en Informatique (LaBRI) et le laboratoire ERIC soutenue par le GDR ISIS (projet jeunes chercheurs).

En segmentation d'images, un paradigme classique repose sur un processus de minimisation d'énergie. L'idée est de déformer une forme géométrique donnée a priori (un modèle) sur l'image analysée en minimisant une énergie, souvent dirigée par deux termes :

- le premier terme correspond à une *énergie image* dirigée par les données. Cette énergie est minimale quand la forme géométrique est localisée sur une zone de l'image *intéressante*. Habituellement, on souhaite que la forme s'adapte aux contours présents dans l'image et donc cette énergie est souvent donnée par l'opposé de la norme du gradient de l'image.
- Le second terme est une énergie liée à la forme de l'objet géométrique. L'idée est que, dans le cas où l'image présente du bruit ou de faibles contrastes, il nous faut *régulariser* cette forme. Classiquement, cette énergie correspond à des mesures de longueur et de courbure de la forme.

Dans la littérature, des courbes paramétriques (*snakes* par exemple) sont utilisées comme formes géométriques à déformer. Le processus de minimisation est itératif, et consiste à déformer cette forme mathématique afin de converger vers une courbe d'énergie localement minimale dans l'image (ce minimum est censé correspondre à la forme recherchée dans l'image).

Ce projet a démarré sur la définition d'un *modèle déformable discret* proposée par LACHAUD et VIALARD (2001). L'idée est de considérer non plus une déformation d'une courbe paramétrique mais directement une déformation d'une courbe ou d'une surface discrète (voir figure A.7). L'intérêt d'une telle approche est double. Dans un premier temps, la forme, dont l'énergie est à minimiser, est dans le même modèle *image* que l'objet à segmenter. De plus, l'ensemble des déformations possibles d'une forme discrète est finie. La recherche d'un minimum local de la fonction d'énergie devient donc un problème combinatoire sur l'image.

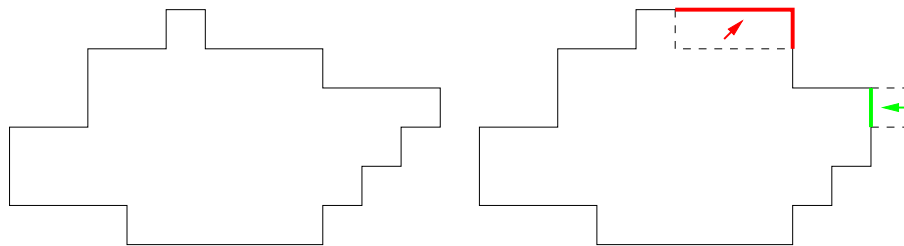


FIG. A.7 –: Exemples de déformation d'un contour discret.

En nous basant sur cette définition de modèle déformable, il nous faut définir la fonction d'énergie associée à ce contour discret. Pour cela, des estimateurs très performants de longueur ou de courbure pour le cas 2D sont nécessaires. Cette collaboration avec le LaBRI a donc pour objectif principal l'intégration des différents estimateurs présentés dans le chapitre 5 dans le processus de calcul d'énergie. Sur le plan expérimental, l'objectif est d'offrir une stabilité et une bonne précision à ce calcul d'énergie tout en utilisant des algorithmes efficaces. Sur le plan théorique, la notion de convergence asymptotique de ces estimateurs s'interprètent directement

dans ce modèle : dans le cas bidimensionnel, si nous utilisons des estimateurs discrets de courbure et de longueur ayant cette propriété de convergence asymptotique, nous pouvons alors montrer que l'énergie d'une courbe discrète converge asymptotiquement vers l'énergie classique d'un *snake* sous-jacent à la courbe discrète, si celui-ci est paramétré par son abscisse curviligne. L'idée de ces recherches est de permettre une mise en correspondance entre modèles déformables discrets et modèles déformables continus.

A.5 Conclusion

Nous avons regroupé dans cette annexe l'ensemble des applications dans lesquelles nous avons utilisé les algorithmes proposés. Ces dernières nous ont servis, en amont, d'inspiration pour le développement des estimateurs du chapitre 5. En aval, ces applications nous ont permis de concrétiser un certain nombre d'avancées théoriques.

Annexe B

Preuves de convergence asymptotique

Sommaire

B.1	Introduction	242
B.2	Courbes en dimension d : approche par <i>tube</i>	242
B.2.1	Estimation de longueur	242
B.2.2	Estimation de normales	246
B.2.3	Estimation de la courbure	247
B.3	Surfaces : aire et normales	247
B.3.1	Estimation de normales surfaciques	248
B.3.2	Estimation d'aire de surfaces discrètes	248

B.1 Introduction

Les différentes preuves présentées dans cette annexe correspondent à une collaboration avec OLIVIER TEYTAUD¹. Ces preuves se basent essentiellement sur des propriétés de géométrie différentielle. Une réécriture de celles-ci dans un formalisme moins complexe est en cours.

B.2 Courbes en dimension d : approche par *tube*

Dans ce paragraphe, nous prouvons la convergence asymptotique des estimateurs de longueur de courbes discrètes 2D et 3D, par segmentation en droites discrètes. Nous prouvons aussi la convergence de l'estimateur de normales 2D basé sur les tangentes de VIALARD (1996a) ainsi que la convergence du calcul de courbure de l'algorithme 27.

Toutes ces preuves s'insèrent dans un même formalisme basé sur une décomposition en *tubes* d'une courbe continue. Dans un premier temps nous présentons la preuve complète des estimateurs de longueur, puis nous dérivons des variantes de ce formalisme pour prouver la convergence de l'estimateur de normales et de courbure.

B.2.1 Estimation de longueur

Par la suite, nous considérons :

1. $\gamma : [0, 1] \rightarrow \mathbb{R}^d$ une courbe continue sans auto-intersection dont la courbure est bornée par $C = \frac{1}{R}$;
2. γ est contenue dans une séquence de tubes T_1, \dots, T_N de diamètre θ et de longueur $L(T_1), \dots, L(T_N)$. Chaque extrémité d'un tube est décomposé comme présenté dans la figure B.1-(a). On appelle ce tube auquel on a ajouté un bord, à chaque extrémité, de longueur $\theta/2$ un θ -*agrandissement* ;
3. T_1 contient $\gamma(0)$;
4. T_N contient $\gamma(1)$;
5. les tubes T_i et T_{i+1} sont adjacents par leur θ -agrandissement. Plus précisément, cette intersection se fait par l'axe principal du tube et les faces extrémités (voir figure B.1-(b)) ;
6. $L(T_i) \geq K \sqrt{\frac{\theta}{C}}$ pour tout i dans $\{1 \dots N\}$ avec $K \in \mathbb{R}$.

L'hypothèse 6 sera discutée plus en détail plus loin. Ce formalisme de tubes correspond à une segmentation en droite discrète de la discrétisation de γ . Dans ce cas, le paramètre θ est lié à la résolution de la grille. Nous supposons θ suffisamment petit devant R pour éviter un certain nombre de cas pathologiques comme des demi-tours dans les tubes (on suppose dans ce cas $\theta/2 < R$) ou encore des cas où un tube T contient deux morceaux de γ (voir figure B.1-(c)).

Par la suite, nous supposerons, sans perte de généralité, que la courbe γ commence en 0 et en termine en 1 dans les parties de longueur $\theta/2$ aux extrémités des tubes T_1 et T_N (voir figure B.1-(b)).

¹. ARTELYS, 215 rue Jean-Jacques Rousseau, 92136 Issy-les-Moulineaux Cedex France, olivier.teytaud@artelys.com

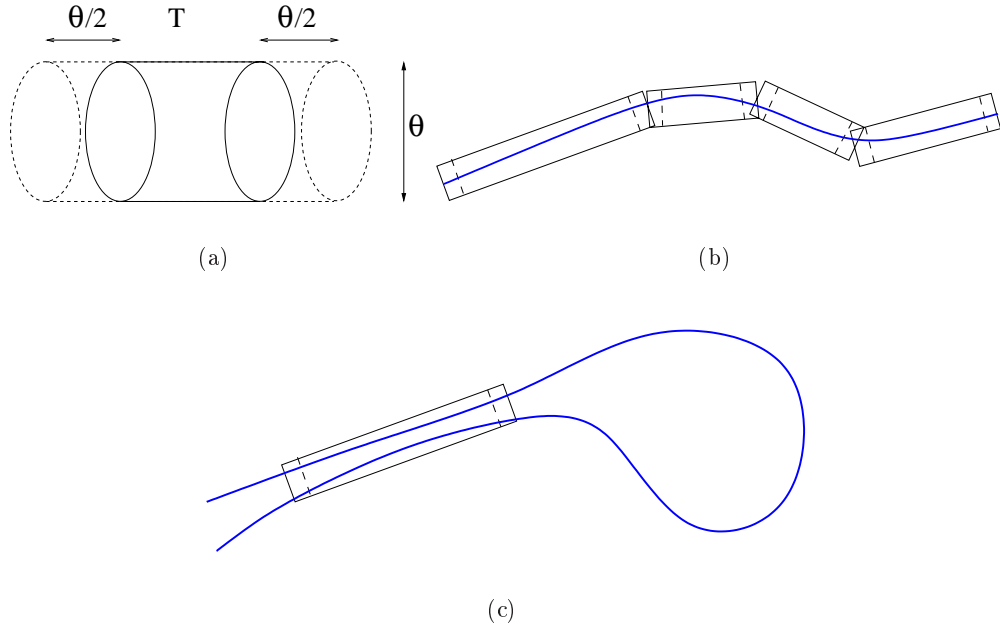


FIG. B.1 –: Illustration du θ -agrandissement d'un tube T (a) et segmentation de γ en une séquence de tubes (b). Le paramètre θ doit être suffisamment petit pour éviter le cas (c).

Si nous notons $\mathcal{L}(\gamma)$ la longueur de γ , l'objectif est de montrer le théorème :

Théorème B.1 Soit γ une courbe vérifiant les hypothèses précédentes et une séquence de tubes $\{T_1, \dots, T_N\}$ (N dépend de θ) vérifiant aussi les propriétés ci-dessus. Nous avons alors :

$$\lim_{\theta \rightarrow 0} \sum_{i=1}^N L(T_i) = \mathcal{L}(\gamma)$$

Dans ce qui suit, nous définissons un θ -rétrécissement de T comme étant une suppression des bords de longueurs $\theta/2$ à un tube T . Nous avons le lemme suivant :

Lemme B.1 L'angle maximal entre l'axe du tube T_i et la tangente en tout point de γ dans le θ -rétrécissement de T_i est inférieur à :

$$\Phi = \arcsin\left(\frac{\theta}{L - \theta}\right) + L(T_i)/R$$

De plus, si θ est suffisamment petit pour que $R \sin(\alpha) \leq L(T_i)$, avec $\alpha = \arctan\sqrt{\frac{\theta}{2R - \theta}}$, alors cet angle maximal Φ est borné par α .

Preuve : Dans le pire cas, nous devons considérer γ comme étant un arc de cercle dans le tube T_i de rayon R . Ainsi, nous avons (voir les notations dans la figure B.2):

$$\begin{aligned} \Omega &\leq L(T_i)/R \\ \sin \beta &\leq \frac{\theta}{L - \theta} \end{aligned}$$

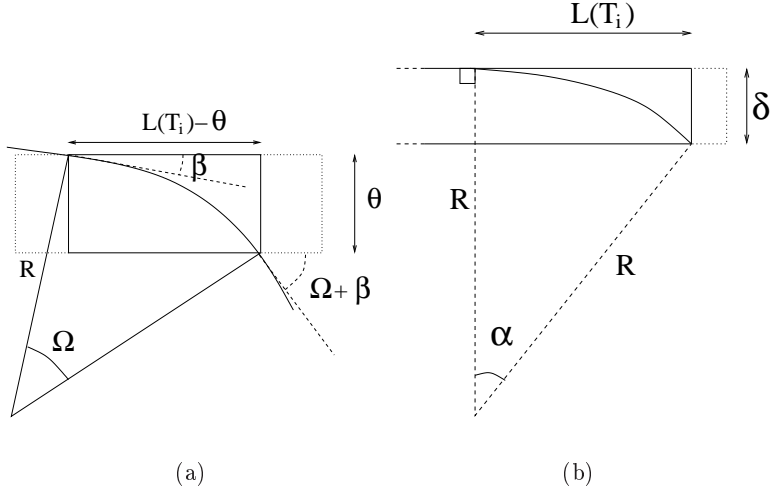


FIG. B.2 –: θ -rétrécissement d'un tube T et différentes notations pour la preuve du lemme B.1. La figure (b) correspond à la projection 2D du pire cas pour le lemme.

L'objectif étant de borner $\Phi = \Omega + \beta$, nous obtenons (voir figure B.2-(a)) :

$$\Phi = \arcsin\left(\frac{\theta}{L - \theta}\right) + L(T_i)/R$$

ce qui prouve le premier résultat.

Le cas particulier selon lequel $R \sin(\alpha) \leq L(T_i)$ pour $\tan \alpha \leq \sqrt{\frac{\theta}{2R - \theta}}$ est présenté figure B.2-(b). Dans ce cas, θ doit être suffisamment petit devant la dérivée de γ pour que γ reste dans le θ -rétrécissement. Finalement, nous obtenons le résultat selon lequel Φ est borné par $\arctan\left(\sqrt{\frac{\theta}{2R - \theta}}\right)$. \square

En corollaire à ce lemme, nous avons une borne en $O(\sqrt{\theta C})$ sur l'angle maximal entre l'axe du tube et la tangente en tout point de γ dans le tube.

De plus, si nous considérons une paramétrisation colinéaire à l'axe du tube de γ , la dérivée de γ dans le tube est bornée par $\sqrt{1 + (d - 1)p^2}$ avec $p = \sin(\Phi)$ et $d : [0, 1]$. Nous obtenons ainsi le corollaire :

Corollaire B.1 La différence entre la longueur de γ dans le tube T_i et la longueur de ce tube est bornée par :

$$L(T_i)(\sqrt{1 + (d - 1)p^2} - 1)$$

Cette erreur est linéaire en $L(T_i)$ et en θC (avec C constant par hypothèse). Ce corollaire prouve donc qu'à l'intérieur d'un tube, l'erreur de l'estimation de longueur converge vers 0 quand θ tend vers 0. Il nous faut maintenant montrer que l'erreur est linéaire en θC à la jonction entre deux tubes.

Nous introduisons le lemme suivant :

Lemme B.2 Si l'angle entre une tangente à γ et l'axe du tube dans le θ -rétrécissement de T_i est borné par Φ , alors cet angle est borné par $\Phi' = \arcsin(\sin(\Phi) + \theta C)$ dans le (θ) -agrandissement de T_i .

Preuve : La figure B.3 présente les différentes notations et la projection 2D du pire cas pour le lemme (arc de cercle de rayon R sortant du tube par une extrémité du θ -rétrécissement).

En utilisant les différentes relations trigonométriques des angles de cette figure, nous avons :

$$R \cos(\Phi') + A = R \cos(\Phi)$$

et

$$R \sin(\Phi') = R \sin(\Phi) + \theta$$

De cette dernière équation, nous trouvons :

$$\Phi' = \arcsin\left(\sin(\Phi) + \frac{\theta}{R}\right)$$

et comme $C = \frac{1}{R}$, nous prouvons le lemme. \square

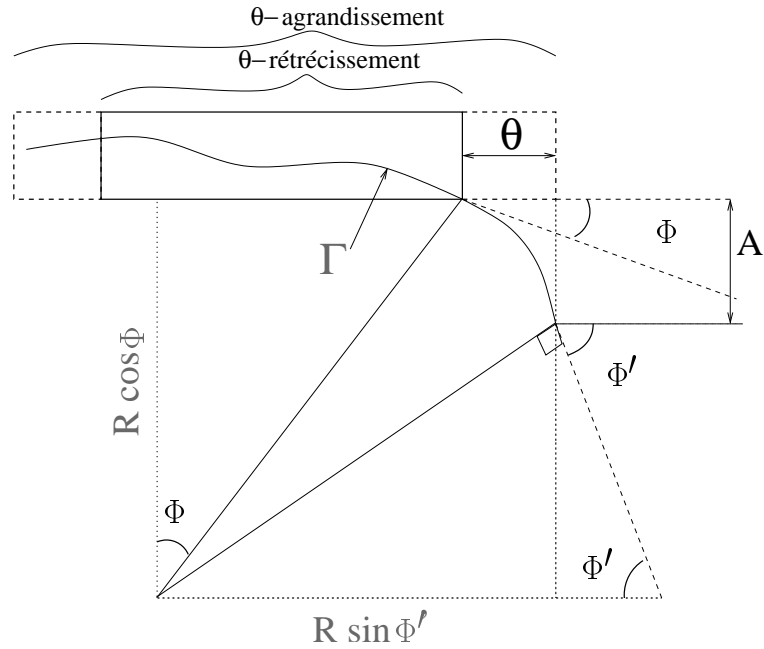


FIG. B.3 –: Projection en 2D du pire cas et illustration des différentes notations pour la preuve du lemme B.2.

Ce lemme indique que γ a une variation d'angle de ses tangentes de Φ' dans le tube T_i , ainsi, l'angle entre deux tubes consécutifs, noté Ψ , est borné par $2\Phi'$. La longueur du morceau de γ entre les deux tubes est donc bornée par :

$$\sqrt{1 + (d-1)p^2} \frac{\Psi\theta}{2}$$

avec $p = \sin(\Phi')$ et $d : [0, 1]$.

Ceci implique le corollaire suivant :

Corollaire B.2 L'erreur de l'estimation de longueur lors de la jonction de deux tubes est bornée par $O(\theta\Psi) = O(\theta^{2/3}\sqrt{C})$. Étant donné que le nombre de jonctions est borné par $O(\sqrt{\frac{C}{\theta}})$, l'erreur sur l'estimation totale de longueur est bornée par $O(\theta C)$

Ce corollaire est obtenu en ajoutant le résultat du corollaire B.1 et du lemme précédent. Pour les extrémités de γ , l'erreur sur l'estimation de longueur en ces deux points est aussi bornée par $O(\theta C)$.

Nous avons donc une estimation de la longueur de γ par la somme des longueurs des tubes qui converge vers $\mathcal{L}(\gamma)$ quand θ tend vers 0. Or, cette structure en tube correspond à une segmentation en droites discrètes de la discrétisation de γ et θ correspond à l'inverse de la résolution de la grille. Nous exprimons ce résultat sous forme du théorème suivant :

Théorème B.2 *Si γ vérifie les hypothèses ci-dessus, l'estimation de longueur basée sur une segmentation de la courbe en segments de droites discrètes en dimension 2 ou en dimension 3 converge asymptotiquement vers la longueur de γ .*

Nous notons que les lemmes et corollaires précédents sont vrais quelle que soit la dimension et en particulier pour les dimensions 2 et 3.

Nous discutons maintenant l'hypothèse selon laquelle $L(T_i) \geq K \sqrt{\theta/C}$, pour tout $i < N$ et $K \in \mathbb{R}$. Dans le cas général, la segmentation en droites discrètes ne vérifie pas cette hypothèse. En effet, des cas pathologiques existent. Par exemple, une courbe γ dont la courbure serait bornée et qui oscillerait autour de l'axe des x avec un intervalle entre les passages par $y = 0$ tendant vers 0. Dans ce cas, l'algorithme de segmentation en droites discrètes ne vérifiera pas cette hypothèse quelle que soit la résolution de la grille. Cependant, la famille de courbes vérifiant cette propriété est très large. Dans le cas 2D par exemple, les courbes convexes vérifient cette propriété ainsi que celles comportant un nombre fini de points d'inflexion.

B.2.2 Estimation de normales

Nous reprenons un formalisme très similaire à ce qui est présenté ci-dessus. Nous considérons :

1. $\gamma : [0, 1] \rightarrow \mathbb{R}^d$ une courbe continue dont la courbure est bornée par $C = \frac{1}{R}$;
2. en un point p de γ , on considère le tube T de longueur $L(T)$ et d'épaisseur θ , centré en p ;
3. $L(T) \geq K \sqrt{\frac{\theta}{C}}$ pour tout i dans $\{1 \dots N\}$ avec $K \in \mathbb{R}$.

Nous conservons les mêmes définitions de θ -agrandissement de T que précédemment. Ce tube correspond à une tangente discrète calculée par l'algorithme de VIALARD (1996a) sur la discrétisation de γ en p , dans le cas 2D.

D'après le lemme B.1, nous savons que l'angle maximal entre la tangente à γ et l'axe du tube T est borné par $O(\sqrt{\theta C})$. Or, cette borne converge vers 0 quand la résolution de la grille augmente, c'est-à-dire quand θ tend vers 0. Ceci nous donne directement le théorème suivant :

Théorème B.3 *Soit une courbe γ dans \mathbb{R}^2 vérifiant les hypothèses ci-dessus, l'estimateur de normales discrètes 2D basé sur l'algorithme de VIALARD (1996a) converge asymptotiquement.*

La discussion sur les hypothèses de ce théorème est identique au cas précédent.

B.2.3 Estimation de la courbure

Nous reprenons le formalisme par tubes avec les définitions suivantes :

- $\gamma : [0, 1] \rightarrow \mathbb{R}^d$ une courbe continue dont la courbure est bornée ;
- soit un point M d'une courbe γ , de courbure $k(M) = \frac{1}{R}$ non nulle. Soit D la droite dont le vecteur directeur est la tangente en M .
- Soient T_1 et T_2 les deux tubes partant de M le long de γ , de part et d'autre de M .
- Soient β_1 et β_2 les angles entre T_1 et D , T_2 et D .
- Nous supposons que $L(T_i) \geq K \sqrt{\frac{\theta}{C}}$ pour tout i dans $\{1, 2\}$ avec $K \in \mathbb{R}$. Nous supposons aussi que $L(T_i) \leq K' \sqrt{\frac{\theta}{C}}$ avec $K' \in \mathbb{R}$. Cette dernière hypothèse est vraie dans le cas des processus de discrétisations standards.

Nous avons alors le théorème suivant :

Théorème B.4 *Soit une courbe γ dans \mathbb{R}^2 ou dans \mathbb{R}^3 vérifiant les hypothèses ci-dessus, l'estimateur de courbure basé sur les demi-tangentes discrètes (algorithme 27) converge asymptotiquement.*

Preuve : *Dans cette preuve, nous considérons le pire cas qui correspond à un arc de cercle de rayon R . Ainsi, d'après le lemme B.1, la longueur du tube T_1 est égale à $R\beta_1$ avec une erreur bornée par $O(\beta_1^2 + \theta)$. De même, la longueur T_2 est égale à $R\beta_2$ avec une erreur en $O(\beta_2^2 + \theta)$.*

En se basant sur ces longueurs, nous avons :

$$L(T_1) + L(T_2) = R\alpha + O(\alpha^2 + \theta)$$

avec $\alpha = \beta_1 + \beta_2$ (voir figure B.4).

De plus, avec l'hypothèse selon laquelle $L(T_i) \leq K' \sqrt{\theta/C}$, α est borné par $O(\sqrt{\theta/R})$.

En d'autres termes, nous estimons la longueur de l'arc de cercle $R\alpha$ par la somme des longueurs des tubes avec une erreur qui tend vers 0 quand θ tend vers 0. Nous utilisons donc ce processus qui converge asymptotiquement pour estimer R en $L(T_1) + L(T_2)$ par $\beta_1 + \beta_2$. Cette estimation est similaire à celle de l'algorithme 27 ce qui prouve sa convergence asymptotique. \square

Cette preuve est valide quelle que soit la dimension de γ et notamment dans les cas 2D et 3D.

B.3 Surfaces : aire et normales

Dans ce paragraphe, nous prouvons la convergence asymptotique de l'estimateur de normales discrètes surfaciques basé sur un algorithme similaire à celui de TELLIER et DEBLED-RENNESON (1999). Nous prouvons ensuite la convergence de l'estimateur d'aire de surface discrète par intégration du champ de normales surfaciques selon le principe proposé par LENOIR (1999).

Nous considérons dans ce paragraphe une surface compacte de Jordan, notée \mathcal{S} , dans \mathbb{R}^3 . Nous considérons la discrétisation de cette surface sur une grille de résolution $r = 1/\theta$.

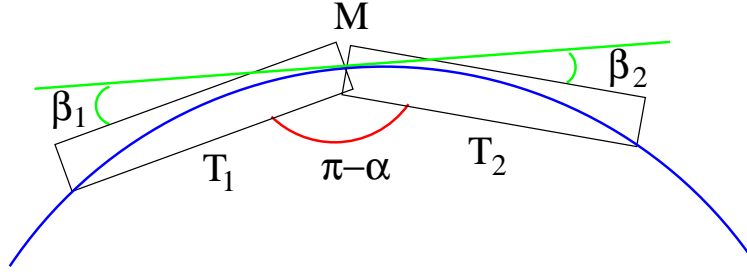


FIG. B.4 –: Illustration pour la preuve du théorème B.4 (projection 2D du pire cas).

B.3.1 Estimation de normales surfaciques

Nous reprenons l'approche de LENOIR (1999) et TELLIER et DEBLED-RENNESON (1999) qui consiste à estimer la normale en un point de la surface par produit vectoriel de deux tangentes calculées sur des coupes de l'objet (voir paragraphe 5.3.2.3).

Ainsi, si nous notons \vec{t}_1^* et \vec{t}_2^* les deux tangentes dans les deux coupes, calculées en un surfel s d'une surface discrète, nous avons :

$$\vec{n}^*(s) = \vec{t}_1^*(s) \times \vec{t}_2^*(s) \quad (\text{B.1})$$

où \times est le produit vectoriel usuel.

D'après les résultats du paragraphe précédent, nous avons le théorème :

Théorème B.5 *L'estimation des normales surfaciques basée sur une décomposition en coupes et basée sur l'algorithme de VIALARD (1996a) pour résoudre le calcul dans les coupes converge asymptotiquement.*

Preuve : Nous supposons que chacune des coupes de \mathcal{S} dans un plan d'axe de la grille donne une courbe γ vérifiant les hypothèses du paragraphe B.2.

Ainsi, nous avons une estimation de $\vec{t}_1^*(s)$ et $\vec{t}_2^*(s)$ qui converge asymptotiquement par le théorème B.3. En d'autres termes, $\vec{t}_1^*(s)$ converge vers la tangente exacte $\vec{t}_1(s)$ en s , de même pour $\vec{t}_2^*(s)$.

Finalement, le produit vectoriel $\vec{t}_1^*(s) \times \vec{t}_2^*(s)$ converge nécessairement vers $\vec{t}_1(s) \times \vec{t}_2(s)$, ce qui prouve le théorème. \square

Encore une fois, l'ensemble des surfaces de Jordan \mathcal{S} vérifiant les hypothèses de ce théorème est vaste. Une étude est en cours pour isoler les surfaces qui ne vérifient pas cet énoncé.

B.3.2 Estimation d'aire de surfaces discrètes

Dans ce paragraphe, nous montrons que l'estimation de l'aire d'une surface discrète basée sur l'intégration du champ des normales converge asymptotiquement si l'estimation des normales surfaciques a cette propriété.

Nous considérons donc une surface \mathcal{S} dans \mathbb{R}^3 avec des dérivées continues. Nous considérons un processus de discrétisation, noté f_r qui est une application de \mathcal{S} dans \mathbb{R}^3 et telle que df_r converge faiblement vers l'identité. En d'autres termes, pour toute fonction continue g , nous avons :

$$\int_{\mathcal{S}} g \cdot df_r(ds) \rightarrow \int_{\mathcal{S}} g ds$$

Nous supposons aussi que df_r et f_r sont bornées pour une résolution suffisamment grande et que f_r converge (sens usuel) vers l'identité. Plus précisément, f_r est une fonction de discrétisation telle que $f_r(\mathcal{S})$ converge vers \mathcal{S} quand r croît.

Ce formalisme n'est pas standard pour représenter un processus de discrétisation. Cependant, tous les processus de discrétisation classiques permettent de construire une telle fonction (non unique). Ce point sera discuté par la suite.

Rappelons que l'aire de la surface \mathcal{S} est donnée par :

$$\int_{\mathcal{S}} \vec{n}(s) d\vec{s}$$

Une approximation discrète de cette formule est donnée par :

$$\int_{f_r(\mathcal{S})} \vec{n}^*(s')$$

avec \vec{n}^* une estimation de la normale surfacique. Pour simplifier les notations, le terme intégrale représente ici une somme finie.

Nous supposons que l'estimation $\vec{n}^*(s')$ converge uniformément vers $\vec{n}(s)$ quand $d(s, s') \rightarrow 0$. Plus précisément, nous avons :

$$\lim_{\epsilon \rightarrow 0} \sup_{d(s, s') \leq \epsilon} |\vec{n}^*(s') - \vec{n}(s)| = 0 \quad (\text{B.2})$$

Nous remarquons que cette hypothèse peut être réécrite comme une simple condition de convergence puisque nous travaillons uniquement sur des fonctions continues sur des ensembles compacts. Cette condition peut donc être vue comme une propriété de convergence asymptotique de l'estimateur de normale surfacique.

L'intégrale discrète est donc égale à :

$$\int_{\mathcal{S}} \vec{n}^*(f_r(s)) df_r(\vec{ds}) \quad (\text{B.3})$$

Si nous notons :

$$\Delta = \left| \int_{\mathcal{S}} \vec{n}^*(f_r(s)) df_r(\vec{ds}) - \int_{\mathcal{S}} \vec{n}(s) d\vec{s} \right| \quad (\text{B.4})$$

et si $1/r$ est suffisamment petit pour que nous ayons $|\vec{n}^*(f_r(s)) - \vec{n}(s)| < \epsilon$, alors le fait que df_r soit borné nous donne :

$$\Delta = O(\epsilon) + \left| \int_{\mathcal{S}} \vec{n}(s) df_r(\vec{ds}) - \int_{\mathcal{S}} \vec{n}(s) d\vec{s} \right| \quad (\text{B.5})$$

Par la propriété de convergence faible de df_r , le terme entre les valeurs absolues $|\cdot|$ converge vers 0 quand r croît. Nous obtenons donc le théorème suivant :

Théorème B.6 *Si df_r converge faiblement vers l'identité quand la résolution augmente, et si $\vec{n}^*(f_r(s))$ converge uniformément vers $\vec{n}(s)$, l'estimation de surface par intégration des normales converge vers l'aire de \mathcal{S} avec une vitesse de convergence en $O(\epsilon)$.*

Détaillons maintenant la construction de la fonction de discrétisation f_r .

Pour $1/r$ suffisamment petit, nous définissons $f_r(s) = s'$ une fonction de \mathcal{S} dans \mathbb{R}^3 telle que $s' = s + k\vec{n}(s)$ ($k \in \mathbb{R}$) et $d(s, s')$ est minimal (voir figure B.5).

L'existence, l'unicité, la continuité et le dérivabilité de f_r sont claires partout sauf aux endroits où \mathcal{S} est perpendiculaire à $f(\mathcal{S})$ (voir figure B.5-(droite)).

Deux approches sont possibles : soit nous ajoutons une hypothèse sur \mathcal{S} pour éviter ce cas (comme dans JONAS et KIRYATI (1997)), soit nous montrons que ce dernier n'arrive jamais avec les hypothèses que nous possédons.

Ce cas pathologique apparaît pour $\vec{n}(s) \in E$ avec E l'ensemble des normales aux normales à la discrétisation (normales élémentaires). E est de mesure nulle puisque E est une union finie d'ensembles de mesures nulles (il n'y qu'un nombre fini de normales à la discrétisation et donc un nombre fini d'hyperplans de normales à ces normales).

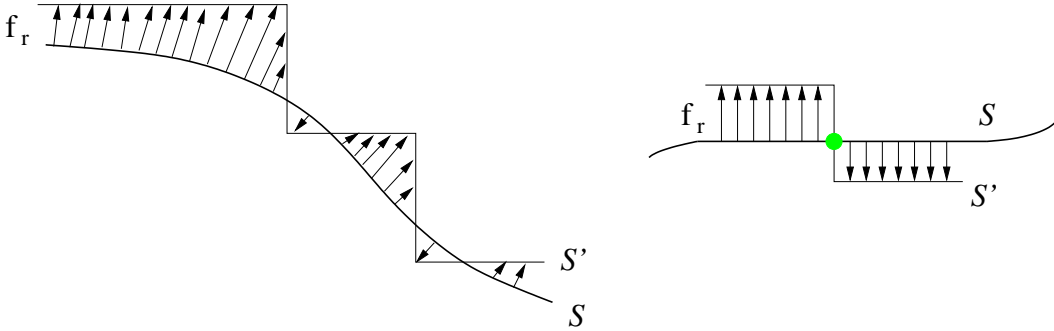


FIG. B.5 –: Illustration de la construction de f_r dans le cas 2D : à gauche une courbe continue \mathcal{S} et sa discrétisation \mathcal{S}' avec la fonction f_r . A droite, présentation d'un cas pathologique : \mathcal{S} est orthogonale à \mathcal{S}' .

Nous avons donc montré que pour un point s donné, la probabilité d'avoir $\vec{n}(s) \in E$ est nulle. Il nous reste à montrer que ceci est vrai pour tout voisinage de $s \in \mathcal{S}$.

Considérons $d(s)$ pour $s \in \mathcal{S}$ la dimension de l'ensemble des normales $\vec{n}(s'')$ pour s'' dans le voisinage de s . Cette fonction d est une fonction à valeurs entières constante partout sauf pour un ensemble de mesure nulle. Nous définissons S_i pour $i \in I$ les différents sous-ensembles connexes maximaux de \mathcal{S} sur lesquels d est différente de la dimension sous-jacente. Nécessairement, I est dénombrable, puisque chaque S_i a une mesure positive, et une somme finie est nécessairement dénombrable.

Ainsi, la somme, pour $i \in I$, des probabilités telles que $\vec{n}(s) \in E$ pour $s \in S_i$ (la variable aléatoire étant le choix des angles définissant la grille par rapport à \mathcal{S}) est une somme dénombrable de probabilités nulles. Finalement, cette probabilité est nulle.

La convergence de f_r vers l'identité étant claire, il nous reste à vérifier la convergence faible de df_r vers l'identité. Ceci est en fait une simple conséquence du fait que f_r converge presque partout vers l'identité : en effet, cette convergence implique une faible convergence de ses dérivées. Précisément, si f_r converge vers l'identité et si la dimension est d , considérons x_i une des variables et g_r (resp. g) la restriction de f_r (resp. l'identité) à une de ses dimensions (on a donc d^2 choix possibles de (x_i, g_r) , correspondant aux 9 cases de la matrice df_r et g est une projection de l'identité sur un axe), et H un hyper-rectangle sur lequel intégrer $\partial g_r / \partial x_i$. L'intégrale de $\partial g_r / \partial x_i$ sur un segment donné par :

$$[(x_1, \dots, x_{i-1}, a_i, x_{i+1}, \dots, x_d), (x_1, \dots, x_{i-1}, b_i, x_{i+1}, \dots, x_d)]$$

est égale à :

$$g_r(x_1, \dots, b_i, \dots, x_d) - g_r(x_1, \dots, a_i, \dots, x_d)$$

Ce dernière mesure tend vers $g(x_1, \dots, b_i, \dots, x_d) - g(x_1, \dots, a_i, \dots, x_d)$. Il suffit alors d'intégrer par rapport aux x_i (le signe intégral et le signe somme commutant pour une famille équicontinue, la famille étant équicontinue car de dérivées bornées) pour voir que l'on a convergence pour les intégrales sur les rectangles, et donc sur les familles continues qui sont limites de fonctions constantes sur des rectangles. Enfin, ceci montre la convergence faible des dérivées de f_r .

Finalement, d'après le théorème B.5, nous obtenons le corollaire suivant :

Corollaire B.3 *Soit \mathcal{S} une surface C^2 dans \mathbb{R}^3 à courbure bornée (plus quelques hypothèses techniques), l'estimateur d'aire de surface discrète basé sur l'intégration discrète des vecteurs normaux, et dont les normales sont données par l'algorithme décrit dans le paragraphe 5.3.2.3, converge asymptotiquement.*

Le théorème précédent est vrai dans le cas de l'estimation de longueur basée sur le même processus d'intégration de normales discrètes. Nous énonçons donc le corollaire :

Corollaire B.4 *Soit Γ la famille des courbes C^2 dans \mathbb{R}^2 à courbure bornée, l'estimateur de longueur basée sur l'intégration des normales discrètes construites par l'algorithme de VIALARD (1996a) converge asymptotiquement.*

Annexe C

Programmation linéaire : algorithme de PREPARATA et SHAMOS

Nous présentons l'algorithme de PREPARATA et SHAMOS (1985) pour le calcul de l'espace des solutions associé à un système d'inéquations linéaires en dimension deux. Cet algorithme, optimal en temps, a été utilisé à de nombreuses reprises pour sa simplicité et son efficacité dans les paragraphes 2.2.2.1, 2.4.2.2 et 3.3.5.

Nous considérons, par la suite, un ensemble \mathcal{E} de n contraintes linéaires (*i.e.* *inéquations linéaires* en dimension 2). Nous supposons que le domaine des solutions $S(\mathcal{E})$ de \mathcal{E} est un polygone fini et non-vide. Ces hypothèses seront traitées au cours de l'analyse.

Pour simplifier l'utilisation de la dualité point/droite, connue aussi sous le nom de *transformée de HOUGH* (version linéaire), nous supposons que \mathcal{E} ne contient aucune contrainte verticale. Si \mathcal{E} présente une telle contrainte, celle-ci pourra être considérée en post-traitement. Sous cette hypothèse, nous définissons le sommet R de $S(\mathcal{E})$ le plus à droite et L le sommet de $S(\mathcal{E})$ le plus à gauche.

Nous notons L^- la courbe polygonale joignant R à L et correspondant à la partie inférieure de $S(\mathcal{E})$. De même, nous notons L^+ la courbe polygonale joignant L à R qui correspond à la partie supérieure de $S(\mathcal{E})$ (voir figure C.1).

Nous représentons par $\{v_i^-\}_{i=0,\dots,m}$ les sommets de L^- , avec $v_0^- = R$ et $v_m^- = L$. De la même manière, nous notons $\{v_i^+\}_{i=0,\dots,m'}$, les sommets de L^+ avec $v_0^+ = L$ et $v_{m'}^+ = R$. Finalement, nous notons aussi les arêtes de L^- par $\{e_i^-\}_{i=1,\dots,m}$ et celles de L^+ par $\{e_i^+\}_{i=1,\dots,m'}$.

Sous notre hypothèse précédente, toutes les contraintes de \mathcal{E} sont données sous la forme :

$$\mathcal{E} = \{a_i x + b_i y + c_i \leq 0\}_{i=0,\dots,n}$$

avec $\{b_i \neq 0\}_{i=0,\dots,n}$.

Dans ce qui suit, nous nous intéressons à la construction de la partie L^+ du polygone des solutions, la partie inférieure L^- se calculant de manière symétrique. Nous remarquons que les arêtes $\{e_i^+\}$ sont construites par le sous-ensemble \mathcal{E}^+ d'inéquations de \mathcal{E} définies par $a_i x + b_i y + c_i \leq 0$ où $b_i > 0$. Plus précisément, si nous considérons l'ensemble des droites issues de \mathcal{E}^+ , les sommets de la courbe polygonale L^+ correspondent aux sommets de l'enveloppe inférieure de ces droites (voir figure C.3).

Pour calculer cette enveloppe inférieure d'un ensemble de droites, nous allons utiliser le passage dans l'espace dual. Si nous notons $y = \alpha x + \beta$, une droite d'une

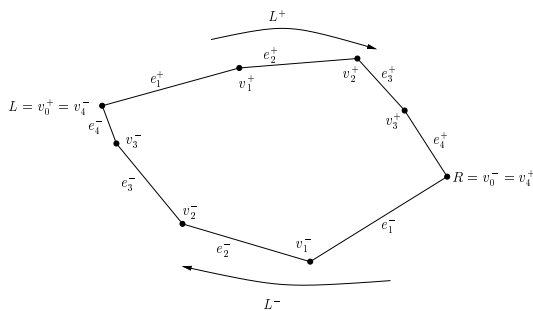


FIG. C.1 –: Différentes notations pour l’illustration de l’algorithme de PREPARATA et SHAMOS (1985).

contrainte de \mathcal{E}^+ (α et β sont donnés par a , b et c), la transformée duale de cette droite correspond au point $(a, -b)$ dans l’espace des paramètres. De la même manière, la transformée de la droite $y = ux - v$ dans l’espace dual est le point (u, v) dans l’espace primal. Nous notons par la suite D^* le point qui correspond à la transformée d’une droite D et P^* la droite correspondant à la transformée d’un point P . La figure C.2 illustre ces différentes définitions.

Nous avons la propriété suivante :

Proposition C.1 Soit $M = (u, v)$ un point dans l’espace primal. M vérifie l’inéquation $ax + by + c \leq 0$, avec $b > 0$, si et seulement si le point $(-a/b, c/b)$ est sous la droite M^* d’équation $y = ux - v$ dans l’espace dual.

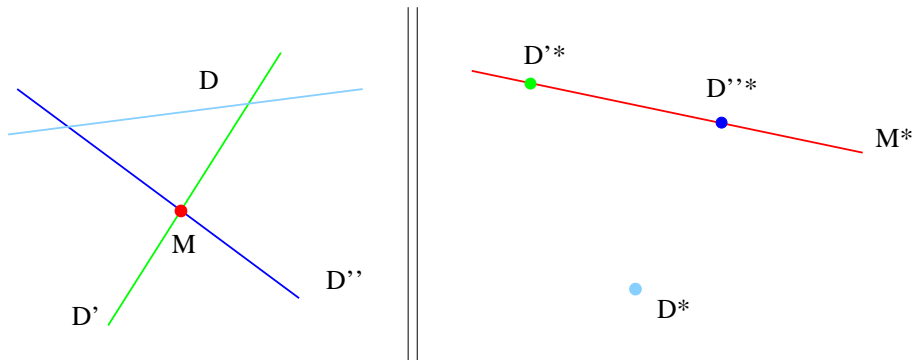


FIG. C.2 –: Transformation dans l’espace dual et illustration de la proposition C.1.

La preuve de cette proposition se base sur le fait que les deux termes de la double implication sont équivalents à l’inégalité $au + bv + c \leq 0$.

Nous pouvons immédiatement déduire de cette proposition que si D, D' et D'' représentent trois droites telles que le point d’intersection $M = D' \cap D''$ a pour coordonnées $M = (a, b)$, alors l’enveloppe inférieure des droites D, D' et D'' prend la valeur b pour $x = a$ si et seulement si D^* est situé sous la droite M^* (voir figure C.2).

En se basant sur cette observation, nous avons :

Proposition C.2 Soit \mathcal{F} un ensemble de droites (dont aucune n’est verticale) et \mathcal{F}^* l’ensemble des points qui sont les transformées des droites de \mathcal{F} dans l’espace dual.

La transformée duale transforme bijectivement les droites de l'enveloppe inférieure de \mathcal{F} en les sommets de la partie supérieure de l'enveloppe convexe de \mathcal{F}^* .

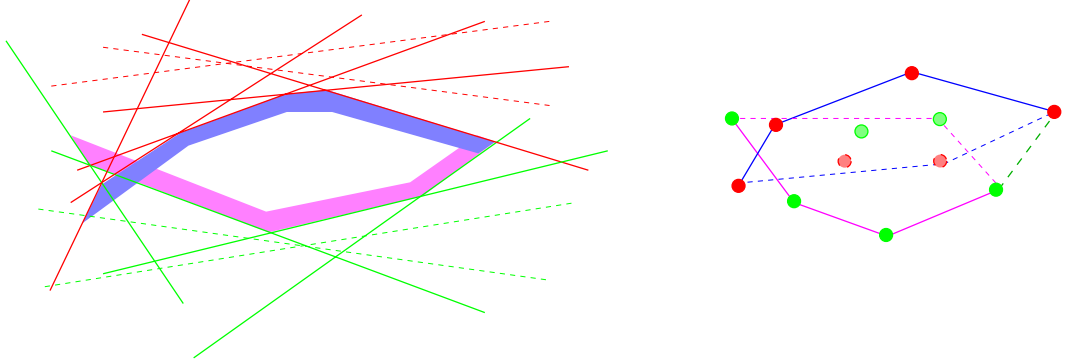


FIG. C.3 – : L'enveloppe inférieure des droites issues de $\mathcal{E}^+ = \{e_i\}$ permettent de construire les sommets de la partie supérieure des points $\{e_i^*\}$. De même pour l'ensemble \mathcal{E}^- .

De plus, les arêtes de l'enveloppe convexe de \mathcal{F}^* correspondent dans le primal aux sommets de l'enveloppe inférieure des droites.

Ainsi, pour calculer L^+ à partir de \mathcal{E}^+ , nous calculons les transformées dans le dual des droites \mathcal{E}^+ , nous appliquons un algorithme classique d'enveloppe convexe des points obtenus et finalement, nous ramenons dans l'espace primal les sommets de la partie supérieure de cette enveloppe convexe.

Si nous avons $|\mathcal{E}^+| = O(n)$, ce calcul s'effectue en $O(n \log(n))$ puisque le calcul de l'enveloppe convexe s'effectue en $O(n \log(n))$.

De la même manière, nous pouvons construire la courbe L^- en considérant les équations $\mathcal{E}^- \subset \mathcal{E}$ données par $a_i x + b_i y + c_i \leq 0$ avec $b_i < 0$. Cette courbe polygonale est l'enveloppe supérieure de ces droites et nous considérons donc la partie inférieure de l'enveloppe convexe des points transformés de \mathcal{E}^- dans l'espace dual. Pour les mêmes raisons que ci-dessus, ce calcul s'effectue en $O(n \log(n))$. Enfin, $S(\mathcal{E})$ est calculé en considérant l'intersection des courbes polygonales L^+ et L^- dans l'espace dual. L'algorithme 31 présente l'algorithme complet.

Corollaire C.1 *Le domaine $S(\mathcal{E})$ des solutions d'un système d'inéquations linéaires en dimension 2 s'obtient en $O(n \log(n))$ en utilisant la transformée duale et le calcul d'enveloppe convexe 2D.*

Nous pouvons maintenant discuter des différents cas où le domaine $S(\mathcal{E})$ est vide ou non borné. Si l'un des deux sous-ensembles \mathcal{E}^+ ou \mathcal{E}^- est vide, l'espace des solutions devient non-borné. Si les deux sous-ensembles sont non vides, $S(\mathcal{E})$ sera non borné si ces deux ensembles contiennent des contraintes du type $b_i y + c_i \leq 0$. Enfin, $S(\mathcal{E})$ sera vide si l'intersection des lignes polygonales L^+ et L^- est vide, ce qui correspond intuitivement au fait que L^- soit *au dessus* de L^+ . Tous ces différents cas peuvent être détectés en pré-traitement de l'algorithme 31 ou au cours de celui-ci sans changer la complexité de l'algorithme.

Étant donné que le coeur de cet algorithme repose sur un calcul d'enveloppe convexe, nous pouvons utiliser des algorithmes classiques pour calculer celle-ci de manière incrémentale et donc rendre la construction de $S(\mathcal{E})$ incrémentale. Nous

Algorithme 31 Calcul de l'intersection de contraintes linéaires par l'algorithme de PREPARATA et SHAMOS (1985)

INTERSECTION_SHAMOS(\mathcal{E})

- 1: $\mathcal{E}^+ = \{a_i x + b_i y + c_i \leq 0 \in \mathcal{E} \mid b > 0\}$
 - 2: $\mathcal{E}^- = \{a_i x + b_i y + c_i \leq 0 \in \mathcal{E} \mid b < 0\}$
 - 3: $C^+ = \text{convexhull}(\text{dual}(\mathcal{E}^+))$
 - 4: $C^- = \text{convexhull}(\text{dual}(\mathcal{E}^-))$
 - 5: Soit L^+ la partie inférieure de C^+
 - 6: Soit L^- la partie supérieure de C^-
 - 7: **retourner intersection de L^+ et de L^-**
-

obtenons donc un coût de mis-à-jour en $O(\log(n))$ à chaque insertion d'une nouvelle contrainte dans la construction de $S(\mathcal{E})$.

Dans le cas d'un système linéaire en dimension 3, le même principe peut être utilisé avec la même complexité, c'est-à-dire $O(n \log(n))$ (pour n inéquations). Nous sommes cependant limités dans les dimensions supérieures par l'algorithme d'enveloppe convexe.

Annexe D

Index des auteurs cités

- Acketa, D. M. 68, 97
Aizawa, K. 80, 83, 84
Amanatides, J. 46
Anderson, T. A. 83, 84
Andrès, E. 17, 46, 56, 57, 81
Aronsson, M. 210
Artzy, E. 12
Asano, T. 181, 183, 186
Attali, D. xiv, 136–138, 141, 143
- Baklouti, F. 234
Baruchel, J. 236
Beckers, A. L. 186, 190
Bernoulli, J. 26
Bertrand, G. 12, 17, 136
Blankinship, W. A. 58
Blum, H. 107, 134
Boissonnat, J.-D. 32, 111
Boix, E. 216
Boller, E. 236, 237
Bolon, P. 127
Borel, S. 236
Borgefors, G. 108, 127
Borianne, P. 71
Braquelaire, J.-P. 40
Bresenham, J. E. xxi, 16, 29, 47, 50, 80, 81, 143
Breu, H. xiv, xxi, 113–119, 121, 124
Bribiesca, E. 14
Brissaud, O. 236
Bruckstein, A. M. 37, 42, 45
Brzoska, J.-B. xvii, 169, 236–238
- Bullard, J. W. xvii, 212–215
Bülow, T. 186, 191
Burguet, J. 72
Buzer, L. 32, 45, 62, 64, 84
- Carter, W. C. xvii, 212–215
Chassery, J.-M. 37, 42, 60
Chehadeh, Y. 127
Chen, L. S. xv, 174
Chin, R. 232
Christoffel, E. B. 27
Cline, H. E. 71
Coeurjolly, D. xxi, 46, 53, 71, 72, 80, 84, 87, 97, 144, 147, 170, 179, 181, 186, 191, 197, 203, 234, 237
Cohen, D. xxi, 46, 47, 49, 50, 167, 168
Coléou, C. xvii, 169, 236–238
Coquin, D. 127
Cormen, T. 116, 143, 150, 151, 217
Couprie, M. 17
Cuisenaire, O. 109, 152
- Damaschke, P. 83, 84
Danielsson, P. E. xiv, 108, 109
de Berg, M. 32, 87, 94, 111, 114, 116, 143, 217
Debled-Rennesson, I. xii, xxi, 19, 28–30, 38–41, 44–49, 51–53, 55, 56, 58–61, 71, 90, 94, 95, 166, 175, 177, 180, 181, 186, 191, 247, 248
Dekker, S. T. 151, 153
Deriche, R. 168, 212
Dorst, L. xi, 22–24, 26, 28, 31, 34, 42, 45, 179, 186

- Ellis, T. J. 183, 186
Ezquerro, N. F. 71
- Ferianc, P. 181
Feschet, F. xv, 166, 167, 177, 184, 204, 206, 209, 234
Figueiredo, O. 47
Fisk, S. 83, 84
Flin, F. 169, 170, 197, 236, 237
François, J. 64, 71, 174
Freeman, H. 13, 21, 178
Freider, G. 12
Fullet, E. R. xvii, 212–215
- Garboczi, E. J. xvii, 212–215
Gerard, Y. xxi, 80, 87, 97, 203
Gil, J. xiv, xxi, 113–119, 121, 124
Gordon, D. 169
Gourdon, A. 212
Grabiner, D. J. xxi, 37, 38, 64
Granum, E. 151
Guan, W. 116
- Hardy, G. H. 27, 34, 40, 90, 91, 95
Harris, J. W. 208
Herman, G. T. xv, 12, 174
Hesselink, W. H. 111, 127
Hirata, T. 111, 127
Hübner, A. 22, 28, 31
Hung, S. H. Y. xi, 22, 23, 28, 56
- Icking, C. 112
Imiya, A. 71
- Jacob, M.-A. 81
Jonas, A. 9, 15, 16, 186, 191, 250
Jordan, C. 10
- Kaufman, A. xxi, 46, 47, 49, 50, 167, 168
Kawamura, Y. 181, 183, 186
Kenmochi, Y. 17, 71
Kim, C. E. xii, 19, 46, 54–56, 61, 80, 83, 84, 93
Kirkpatrick, D. xiv, xxi, 113–119, 121, 124
- Kiryati, N. 9, 15, 16, 108, 155, 186, 191, 250
Klein, R. 112
Klette, R. xv, 22, 28, 31, 165, 179–181, 183, 185, 186, 191, 194, 195
Köllner, P. 112
Koplowitz, J. 23
Kovalevsky, V. A. xiv, xv, 12, 13, 24, 79, 84, 99, 101, 165, 180, 181, 185, 186
Kubler, O. 108
- Lachaud, J.-O. 71, 239
Lamboley, P. 169
Leiserson, C. 116, 143, 150, 151, 217
Lengagne, R. 168, 212
Lenoir, A. 143, 175, 177, 183, 195, 196, 213, 215, 247, 248
Lesaffre, B. xvii, 169, 236–238
Lindblad, J. 194
Lindenbaum, M. 23, 37, 42, 45
Lohou, C. 136
Lorensen, W. E. 71
Ludwig, W. 236, 237
- Ma, L. 112
Ma, S. 116
Macq, B. 109
Malgouyres, R. 12, 72, 175, 177
Maurer Jr., C.R. 158
McIlroy, M. D. xii, 34–36
Megiddo, N. 31, 32, 45, 62, 83
Meijster, A. 111, 127
Mesmoudi, M. M. 61
Miguet, S. 71, 144, 147, 203
Monga, O. 168, 212
Montanari, U. 136
Montanvert, A. 8, 14, 16, 203, 208
Moreau, P. 151, 153
Morgenthaler, D. G. 12
Mullikin, J. C. 109, 194
Nakamura, A. 80, 83, 84
Nehlig, P. 46
Nicod, J.-M. 71
Nyström, I. 194

Obokata, K. 181, 183, 186
 O'Rourke, J. xii, 33, 45
 Overmars, M. 32, 87, 94, 111, 114, 116, 143, 217

 Papier, L. xv, 71, 72, 174, 175
 Perroton, L. 12, 13
 Pfaltz, J. L. 107, 108, 134
 Pieritz, R. A. xvii, 237, 238
 Piper, J. 151
 Pitteway, M. L. V. xxi, 81, 82
 Preparata, F. P. x, xviii, xxii, 33, 38, 45, 64, 65, 84, 87, 94, 96–98, 101, 111, 229, 253, 254, 256
 Proffitt, D. 183, 186

 Qi, R. 158

 Raghavan, V. 158
 Ragnemalm, I. 109
 Remy, E. 108, 134, 136
 Rémy, J. L. 94, 95
 Reveillès, J.-P. xii, xxi, 1, 23–25, 27, 29, 30, 38, 44, 46, 47, 50, 51, 58, 66, 80, 87, 97, 180, 186, 203
 Revenu, M. 175, 177
 Reynolds, R. A. xv, 169, 174
 Rivest, R. 116, 143, 150, 151, 217
 Roerdink, J. B. T. M. 111, 127
 Rosen, D. 183, 186
 Rosenfeld, A. 10, 12, 21, 22, 28, 31, 54, 93, 107, 108, 134
 Rouyer-Degli, J. 94, 95
 Rutkowski, W. 183, 186

 Saito, T. xiv, 105, 109–111, 117, 124, 127
 Salam, I. 46
 Sanniti di Baja, G. 155
 Sarrut, D. 71
 Sauer, P. 83, 84
 Schramm, J. M. 64, 71
 Schwarzkopf, O. 32, 87, 94, 111, 114, 116, 143, 217
 Sethian, J. A. 159
 Shamos, M. I. x, xviii, xxii, 33, 38, 45, 64, 65, 84, 87, 94, 96–98, 101, 111, 229, 253, 254, 256
 Shimony, E. 46
 Sintorn, I.-M. 108, 127
 Sivignon, I. 71, 72
 Sloboda, F. 181, 186, 194, 195
 Smeulders, A. W. M. xi, 22–24, 26, 28, 31, 34, 42, 45, 83, 84, 179, 186, 190, 203, 204
 Soille, P. 143, 152, 156, 159
 Srihari, S. N. 12
 Stocker, H. 208
 Stojmenović, I. 46, 61
 Sun, H. J. 194
 Svensson, S. 108, 155
 Székely, G. 155

 Tajine, M. 64, 71
 Teh, C.-H. 232
 Tellier, P. 175, 177, 247, 248
 Teytaud, O. 46, 53, 170, 181, 186, 191, 197, 237
 Thiel, E. 108, 143
 Thirion, J. P. 212
 Thomassen, C. 73
 Thürmer, G. 168, 169, 175
 Toriwaki, J. I. xiv, 105, 109–111, 117, 124, 127
 Tosić, R. 46
 Tougne, L. xv, xxi, 71, 80, 82, 87, 97, 144, 147, 166, 167, 170, 177, 184, 197, 203, 204, 206, 209, 234, 237
 Troesch, A. 165

 Udupa, J. K. xv, 12, 13, 174

 van der Waerden, B. L. 164
 van Kreveld, M. 32, 87, 94, 111, 114, 116, 143, 217
 Veelaert, P. xii, 19, 56, 57, 61
 Verbeek, P. W. 151, 153, 194
 Verwer, B. J. H. 151, 153
 Vialard, A. xvii, 21, 28, 39–42, 165, 166, 181, 204, 206–208, 239, 242, 246, 248, 251
 Vidal, B. 46

- Vittone, J. xxi, 35, 37, 38, 42, 45, 58, 60, 62, 64, 71
Voss, K. 22, 28, 31, 94, 95
Vossepoel, A. M. 179
Werman, M. xiv, xxi, 113–119, 121, 124
West, D. B. 72, 73
Woo, A. 46
Worring, M. 83, 84, 203, 204
Wright, E. M. 27, 34, 40, 90, 91, 95
Wu, L. D. 22, 28
Wüthrich, C. A. 168, 169, 175
Xu, K. xvii, 237, 238
Yaacoub, G. 66
Yagel, R. 167, 168
Yamada, H. 109
Yip, B. xv, 165, 181, 185, 186
Yu, D. 195
Yvinec, M. 32, 111
Zatko, B. 181, 186, 194, 195
Žunić, J. D. 68, 97, 180

Bibliographie

- ACKETA, D. M. et ŽUNIĆ, J. D. «On the maximal number of edges of convex digital polygons included into an $m \times m$ -grid». *Journal of Combinatorial Theory*, Serie A(69):358–368. **1995**.
- AMANATIDES, J. et WOO, A. «A fast voxel traversal algorithm for ray tracing». Dans *Eurographic's 87*, pages 3–12. **1987**.
- ANDRÈS, E. «Le plan discret». Dans *Colloque de géométrie discrète en imagerie : fondements et applications*. **1993**.
- ANDRÈS, E. *Cercles Discrets et Rotations discrètes*. Thèse de doctorat, Université Louis Pasteur - Strasbourg. **1994**.
- ANDRÈS, E. «Discrete circles, rings and spheres». *Computers and Graphics*, 18(5):695–706. **1994**.
- ANDRÈS, E. *Modélisation analytique discrète d'objets géométriques*. Habilitation à diriger les recherches, Laboratoire IRCOM-SIC, Université de Poitiers. **2000**.
- ANDRÈS, E. «Defining Discrete Objects for Polygonalization: the Standard Model». Dans *10th International Conference on Discrete Geometry for Computer Imagery*, numéro 2301 dans LNCS, pages 313–325. Springer. **2002**.
- ANDRÈS, E. et JACOB, M.-A. «The Discrete Analytical Hyperspheres:». *IEEE Transactions on Visualization and Computer Graphics*, 3(1):75–86. **1997**.
- ARONSSON, M. «Estimating fibre twist and aspect ratios in 3D voxel volumes». Dans *International Conference on Pattern Recognition*. IEEE Computer Society Press, Quebec, Canada. **2002**.
- ARTZY, E., FREIDER, G. et HERMAN, G. T. «The Theory, Design, Implementation and Evaluation of a Three-dimensional Surface Detection Algorithm». Dans *Computer Graphics and Image Processing*, pages 1–24. **1981**.
- ASANO, T., KAWAMURA, Y., KLETTE, R. et OBOKATA, K. «Minimum-Length Polygons in approximation sausages». Dans C. Arcelli, L. Cordella et G. Sanniti di Baja, éditeurs, *International Workshop on Visual Form 4*, numéro 2059 dans LNCS, pages 103–112. Springer Verlag. Capri, Italy. **2001**.
- ATTALI, D. *Squelettes et graphes de Voronoi 2D et 3D*. Thèse de doctorat, Université Joseph Fourier - Grenoble 1. **1995**.
- ATTALI, D., SANNITI DI BAJA, G. et THIEL, E. «Skeleton simplification through non significant branch removal». *Image Processing and Communications*, 3(3-4):63–72. **1997**.

- BAKLOUTI, F. *Classification de courbes 2D appliquée à des profils de stèles*. Laboratoire ERIC, université Lumière Lyon 2. **2002**.
- BECKERS, A. L. et SMEULDERS, A. W. M. «Optimization of Length Measurements for Isotropic Distance Transformations in Three Dimensions». *Computer Vision, Graphics, and Image Processing: Image Understanding*, 55(2):296–306. **1992**.
- BERNOULLI, J. *Recueil pour les astronomes*, chapitre Sur une nouvelle espee de calcul, pages 255–284. **1771**.
- BERTRAND, G. et MALGOUYRES, R. «Some Topological Properties of Discrete Surfaces». Dans *6th International Workshop on Discrete Geometry for Computer Imagery*, numéro 1176 dans LNCS, pages 325–336. Springer-Verlag. **1996**.
- BLANKINSHIP, W. A. «A new version of the Euclidean algorithm». *American Mathematics Monthly*, pages 742–745. **1963**.
- BLUM, H. «A transformation for extracting descriptors of shape». Dans *Models for the Perception of Speech and Visual Forms*, pages 362–380. MIT Press. **1967**.
- BOISSONNAT, J.-D. et YVINEC, M. *Géométrie algorithmique*. Ediscience International. **1995**.
- BOIX, E. *Approximation linéaire des surfaces de \mathbb{R}^3 et applications*. Thèse de doctorat, École Polytechnique, Paris. **1995**.
- BORGEFORS, G. «Distance Transformations in Digital Images». *Computer Vision, Graphics, and Image Processing*, 34(3):344–371. **1986**.
- BORGEFORS, G. et SVENSSON, S. «Optimal Local Distances for Distance Transforms in 3D Using an Extended Neighbourhood». Dans *International Workshop on Visual Form 4*, numéro 2059 dans LNCS, pages 113–122. Springer Verlag. **2001**.
- BORIANNE, P. et FRANÇON, J. «Reversible polyhedrization of discrete volumes». Dans *4th Discrete Geometry for Computer Imagery*, pages 157–168. Grenoble, France. **1994**.
- BRAQUELAIRE, J.-P. et VIALARD, A. «Euclidean Paths: A New Representation of Boundary of Discrete Regions». *Graphical Models and Image Processing*, 61(1):16–43. **1999**.
- BRESENHAM, J. E. «Algorithm for computer control of a digital plotter». Dans *IBM System Journal*, volume 4, pages 25–30. **1965**.
- BRESENHAM, J. E. «A Linear Algorithm for Incremental Digital Display of Circular Arcs». *Communications of the ACM*, 20(2):100–106. **1977**.
- BREU, H., GIL, J., KIRKPATRICK, D. et WERMAN, M. «Linear Time Euclidean Distance Transform Algorithms». *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(5):529–533. **1995**.
- BRIBIESCA, E. «A new chain code». *Pattern Recognition*, 32:235–251. **1999**.
- BRZOSKA, J.-B. *Etudes tridimensionnelles sur la microstructure de la neige*. Habilitation à diriger les recherches, Université Joseph Fourier, Grenoble, Météo-France/CNRM, Centre d'études de la neige. **2001**.

- BRZOSKA, J.-B., COLÉOU, C., LESAFFRE, B., BOREL, S., BRISSAUD, O., LUDWIG, W., BOLLER, E. et BARUCHEL, J. «3D visualization of snow samples by microtomography at low temperature». *ESRF Newsletter*, 32:22–23. **1999a**.
- BRZOSKA, J.-B., LESAFFRE, B., COLÉOU, C., XU, K. et PIERITZ, R. A. «Computation of 3D curvature on a wet snow sample». *The European Physical Journal Applied Physics*. **1999b**.
- BULLARD, J. W., GARBOCZI, E. J., CARTER, W. C. et FULLET, E. R. «Numerical methods for computing interfacial mean curvature». *Computational materials science*, 4:103–116. **1995**.
- BURGUET, J. et MALGOUYRES, R. «Strong thinning and polyhedrisation of the surface of a voxel object». Dans *9th Discrete Geometry for Computer Imagery*, pages 222–234. LNCS 1953, Springer-Verlag. **2000**.
- BUZER, L. «An Incremental Linear Time Algorithm for Digital Line and Plane Recognition Using a Linear Incremental Feasibility Problem». Dans *10th International Conference, Discrete Geometry for Computer Imagery*, numéro 2301 dans LNCS, pages 372–381. Springer Verlag, Bordeaux, France. **2002**.
- CHASSERY, J. M. et MONTANVERT, A. *Géométrie discrète en analyse d'images*. Éditions Hermes. **1991**.
- CHEN, L. S., HERMAN, G. T., REYNOLDS, R. A. et UDUPA, J. K. «Surface shading in the Cuberille environment». *IEEE Journal of computer graphics and Application*, 5(12):33–43. **1985**.
- CHRISTOFFEL, E. B. «Observatio Arithmetica». *Annali di Mathematica*, 6:148–152. 2nd series. **1875**.
- COEURJOLLY, D. «Courbure d'objets discrets». Rapport de DEA, École doctorale IIS, Lyon. Stage effectué au laboratoire ERIC, Université Lumière Lyon 2. **2000**.
- COEURJOLLY, D. «Visibility in Discrete Geometry: An Application to Discrete Geodesic Paths». Dans A. V. A. Braquelaire, J.-O. Lachaud, éditeur, *10th International Conference on Discrete Geometry for Computer Imagery*, pages 326–327. Springer-Verlag, LNCS, 2301, Bordeaux, France. **2002**.
- COEURJOLLY, D., DEBLED-RENNESON, I. et TEYTAUD, O. «Segmentation and Length Estimation of 3D Discrete Curves». Dans G. Bertrand, A. Imiya et R. Klette, éditeurs, *Digital and Image Geometry*, pages 295–313. Springer-Verlag LNCS, 2243. **2001a**.
- COEURJOLLY, D., FESCHET, F. et TOUGNE, L. «Extraction de primitives géométriques pour la classification automatique de profils de stèles funéraires». Dans *8ièmes Rencontres de la Société Francophone de Classification*. Guadeloupe, France. **2001b**.
- COEURJOLLY, D., FLIN, F., TEYTAUD, O. et TOUGNE, L. «Multigrid Convergence and Surface Area Estimation». Dans *Theoretical Foundations of Computer Vision, Geometry, Morphology, and Computational Imaging*. Springer Verlag, LNCS. A paraître. **2002a**.

- COEURJOLLY, D., GERARD, Y., REVEILLÈS, J.-P. et TOUGNE, L. «An elementary algorithm for digital arc segmentation». Dans S. Fourey, G. T. Herman et T. Y. Kong, éditeurs, *International Workshop on Combinatorial Image Analysis*, volume 46 de *Electronic Notes in Theoretical Computer Science*. Elsevier Science Publishers, Temple University, Philadelphia, U.S.A. **2001c**.
- COEURJOLLY, D. et KLETTE, R. «A Comparative Evaluation of Length Estimators». Dans *International Conference on Pattern Recognition*. IEEE Computer Society Press, Quebec. **2002**.
- COEURJOLLY, D., MIGUET, S. et TOUGNE, L. «Discrete Curvature based on Osculating Circle Estimation». Dans C. Arcelli, L. Cordella et G. Sanniti di Baja, éditeurs, *International Workshop on Visual Form 4*, numéro 2059, pages 303–312. Springer-Verlag, LNCS, 2059. Capri, Italy. **2001d**.
- COEURJOLLY, D., MIGUET, S. et TOUGNE, L. «2D and 3D Visibility in Discrete Geometry: an Application to Discrete Geodesic Paths». *Pattern Recognition Letters*. À paraître. **2002b**.
- COEURJOLLY, D., SARRUT, D. et TOUGNE, L. «Décimation en Imagerie Médicale 3D». Dans *Courbes Surfaces et Algorithmes*. Journées du Groupe de Travail «Modélisation Géométrique», Grenoble, France. **1999**.
- COHEN, D. et KAUFMAN, A. «Scan-Conversion Algorithms for linear and Quadratic Objects». Dans *IEEE Computer Society Press, Los Alamitos, Calif*, pages 280–301. **1991**.
- COLÉOU, C., LESAFFRE, B., BRZOSKA, J.-B., LUDWIG, W. et BOLLER, E. «Three-dimensional snow images by X-ray microtomography». *Annals of Glaciology*, 31:75–81. **2001**.
- COQUIN, D., CHEHADEH, Y. et BOLON, P. «3D local distance operator on parallelepipedic grids». Dans *4th Discrete Geometry for Computer Imagery*, pages 147–156. Clermont-Ferrand, France. **1994**.
- CORMEN, T., LEISERSON, C. et RIVEST, R. *Introduction à l'algorithmique*. Dunod. **1990**.
- COUPRIE, M., BERTRAND, G. et KENMOCHI, Y. «Discretization in 2D and 3D Orders». Dans *10th International Conference on Discrete Geometry for Computer Imagery*, numéro 2301 dans LNCS, pages 301–312. Springer. **2002**.
- CUISINAIRE, O. *Distance Transformations : Fast Algorithms and Applications to Medical Image Processing*. Thèse de doctorat, Université Catholique de Louvain. **1999**.
- CUISINAIRE, O. et MACQ, B. «Fast and exact signed Euclidean distance transformation with linear complexity». Dans *IEEE International Conference on Acoustics, Speech and Signal Processing*. **1999**.
- DAMASCHKE, P. «The linear time recognition of digital arcs». *Pattern Recognition Letters*, (16):543–548. **1995**.
- DANIELSSON, P. E. «Euclidean Distance Mapping». *Computer Graphics and Image Processing*, 14(3):227–248. **1980**.

- DE BERG, M., VAN KREVELD, M., OVERMARS, M. et SCHWARZKOPF, O. *Computational Geometry*. Springer-Verlag. **2000**.
- DEBLED-RENNESON, I. *Etude et reconnaissance des droites et plans discrets*. Thèse de doctorat, Université Louis Pasteur. **1995**.
- DEBLED-RENNESON, I., RÉMY, J. L. et ROUYER-DEGLI, J. «Detection of discrete convexity of polyominoes». Dans G. Borgefors, I. Nyström et G. Sanniti di Baja, éditeurs, *9th International Conference in Discrete Geometry for Computer Imagery*, pages 491–504. Uppsala, Sewden. **2000**.
- DEBLED-RENNESON, I. et REVEILLÈS, J.-P. «A linear algorithm for segmentation of digital curves». Dans *International Journal on Pattern Recognition and Artificial Intelligence*, volume 9, pages 635–662. **1995**.
- DORST, L. et SMEULDERS, A. W. M. «Discrete Representation of Straight Lines». *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:450–463. **1984**.
- DORST, L. et SMEULDERS, A. W. M. «Length Estimators for Digitized Contours». *Computer Vision, Graphics, and Image Processing*, 40(3):311–333. **1987**.
- DORST, L. et SMEULDERS, A. W. M. «Decomposition of discrete curves into piecewise straight segments in linear time». Dans *Contemporary Mathematics*, volume 119. **1991a**.
- DORST, L. et SMEULDERS, A. W. M. «Discrete Straight Line Segments: Parameters, Primitives and Properties». Dans R. Melter, P. Bhattacharya et A. Rosenfeld, éditeurs, *Vision Geometry, series Contemporary Mathematics*, volume 119, pages 45–62. American Mathematical Society. **1991b**.
- ELLIS, T. J., PROFFITT, D., ROSEN, D. et RUTKOWSKI, W. «Measurement of the Lengths of Digitized Curved Lines». *Computer Graphics and Image Processing*, 10(4):333–347. **1979**.
- FESCHET, F. et TOUGNE, L. «Optimal Time Computation of the Tangent of a Discrete Curve : Application to the Curvature». Dans *8th International Workshop in Discrete Geometry for Computer Imagery*, pages 31–40. Springer-Verlag, LNCS, 1568. **1999**.
- FIGUEIREDO, O. et REVEILLÈS, J.-P. «A Contribution to 3D Digital Lines». Dans *5th International Workshop in Discrete Geometry for Computer Imagery*. Clermont-Ferrand - France. **1995**.
- FISK, S. «Separating Points Sets by Circles, and the Recognition of Digital Disks». *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 8(4):554–556. **1986**.
- FLIN, F. *Imagerie 3D de neige par tomographie X : méthode de détermination des vecteurs normaux à une surface discrète*. Mémoire de DEA, Météo-France/CNRM, Centre d'études de la neige. **2000**.
- FLIN, F., BRZOSKA, J.-B., LESAFFRE, B., COLÉOU, C. et LAMBOLEY, P. «Computation of normal Vectors of discrete 3D objects: application to natural snow images from X-ray tomography». *Image Anal. Stereol.*, 20:187–191. **2001**.

- FLIN, F., BRZOSKA, J.-B., LESAFFRE, B., COLÉOU, C. et PIERITZ, R. A. «Full 3D modeling of curvature-dependant snow metamorphism: first results and comparison with experimental tomographic data». Submitted to *Journal of Physics D: Applied Physics*. **2002**.
- FRANÇON, J. et PAPIER, L. «Polyhedrization of the Boundary of a Voxel Object». Dans *8th International Workshop in Discrete Geometry for Computer Imagery*, pages 425–434. Springer-Verlag, LNCS, 1568. **1999**.
- FRANÇON, J., SCHRAMM, J. M. et TAJINE, M. «Recognizing arithmetic straight lines and planes». Dans *6th Discrete Geometry for Computer Imagery*, pages 141–150. Springer-Verlag, LNCS, 1176. **1996**.
- FREEMAN, H. «On Encoding Arbitrary Geometric Configurations». *IRE Transactions on Electronic Computers*, 10:260–268. **1961**.
- FREEMAN, H. «Computer Processing of Line-Drawing Images». *ACM Computing Surveys*, 6(1):57–97. **1974**.
- GORDON, D. et REYNOLDS, R. A. «Image-space shading of 3-dimensional objects». *Computer Vision, Graphics and Image Processing*, 29:361–376. Reprinted in *Volume Visualization*, A. Kaufman, ed., IEEE Computer Society Press (1991) 185–200. **1985**.
- GRABINER, D. J. «Farey nets and multidimensional continued fractions». *Monasth. Math.*, 114(4):35–61. **1992**.
- GUAN, W. et MA, S. «A List-Processing Approach to Compute Voronoi Diagrams and the Euclidean Distance Transform». *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20:757–761. **1998**.
- HARDY, G. H. et WRIGHT, E. M. *Introduction to the theory of numbers*. Oxford press. **1975**.
- HARRIS, J. W. et STOCKER, H. *Handbook of mathematics and computational science*. Springer-Verlag. **1998**.
- HIRATA, T. «A unified linear-time algorithm for computing distance maps». *Information Processing Letters*, 58(3):129–133. **1996**.
- HOCHBAUM, D. S., éditeur. *Approximation algorithms for NP-hard problems*. PWS Publishing Company. **1997**.
- HÜBLER, A., KLETTE, R. et VOSS, K. «Determination of the convex hull of a finite set of planar points within linear time». *EIK (Elektronische Informationsverarb. Kybernetik)*, 17:121–140. **1981**.
- HUNG, S. H. Y. «On the Straightness of Digital Arcs». *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7:203–215. **1985**.
- ICKING, C., KLEIN, R., KÖLLNER, P. et MA, L. «VoroGlide». URL <http://wwwpi6.fernuni-hagen.de/Geometrie-Labor/VoroGlide>, version 2.2. **2001**.
- JONAS, A. et KIRYATI, N. «Digital representation schemes for 3D curves». *Pattern Recognition*, 30(11):1803–1816. **1997**.
- JONAS, A. et KIRYATI, N. «Length estimation in 3-D using cube quantization». *Journal of Mathematical Imaging and Vision*, 8:215–238. **1998**.

- JORDAN, C. *Cours d'analyse III*. Paris. **1887**.
- KAUFMAN, A. «Efficient algorithms for 3D scan-conversion of parametric curves, surfaces, and volumes». *Computer Graphics*, 21(4):171–179. **1987**.
- KAUFMAN, A. et SHIMONY, E. «3-D scan conversion algorithms for voxel-based graphics». Dans *ACM Workshop on Interactive 3D Graphics*, ACM Press, NY, pages 45–75. **1986**.
- KENMOCHI, Y., IMIYA, A. et EZQUERRA, N. F. «Polyhedra generation from lattice points». Dans *6th Discrete Geometry for Computer Imagery*, pages 127–138. LNCS 1176, Springer Verlag. **1996**.
- KIM, C. E. «On the cellular convexity of complexes». *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-3:617–625. **1981**.
- KIM, C. E. «Three-dimensional digital line segments». Dans *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 5, pages 231–234. **1983**.
- KIM, C. E. «Digital Disks». *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:372–374. **1984a**.
- KIM, C. E. «Three-Dimensional Digital Planes». *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:639–645. **1984b**.
- KIM, C. E. et ANDERSON, T. A. «Digital Disks and a Digital Compactness Measure». Dans *Seventh International Conference on Pattern Recognition*, IEEE Publ. 84CH2046-1, pages 254–257. IEEE. **1984**.
- KIM, C. E. et ROSENFELD, A. «Convex digital solids». *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 4(6):612–618. **1982**.
- KIM, C. E. et STOJMENOVIĆ, I. «On the recognition of digital planes in three-dimensional space». *Pattern Recognition Letters*, (12):665–669. **1991**.
- KIRYATI, N. et KUBLER, O. «On Chain Code Probabilities and Length Estimators for Digitized Three Dimensional Curves». **1995**.
- KIRYATI, N. et SZÉKELY, G. «Estimating shortest paths and minimal distances on digitized three-dimension surfaces». *Pattern Recognition*, 26(11):1623–1637. **1993**.
- KLETTE, R. et BÜLOW, T. «Minimum-Length Polygons in Simple Cube-Curves». Dans *Discrete Geometry for Computer Imagery*, pages 467–478. Springer-Verlag, LNCS, 1953. **2000**.
- KLETTE, R., KOVALEVSKY, V. A. et YIP, B. «Lenth Estimation of Digital Curves». Dans *Vision Geometry VIII - SPIE*, pages 117–129. Denver. **1999**.
- KLETTE, R. et SUN, H. J. «Digital planar segment based polyhedrization for surface area estimation». Dans C. Arcelli, L. P. Cordella et G. Sanniti di Baja, éditeurs, *International Workshop on Visual Form 4*, volume 2059 de *Lect. Notes Comput. Sci.*, pages 356–366. Springer-Verlag. **2001**.
- KLETTE, R. et ŽUNIĆ, J. D. «Convergence of Calculated Features in Image Analysis». (CITR-TR-52). **1999**.

- KOVALEVSKY, V. A. «New Definition and Fast Recognition of Digital Straight Segments and Arcs». *Proceedings of the tenth international conference on Pattern Analysis and Machine Intelligence*. **1990**.
- KOVALEVSKY, V. A. «A Topological Method of Surface Representation». *Discrete Geometry for Computer Imagery*, pages 118–135. **1999**.
- LACHAUD, J.-O. «Topologically defined isosurfaces». Dans *6th Discrete Geometry for Computer Imagery*, pages 245–256. LNCS 1176, Springer-Verlag. **1996**.
- LACHAUD, J.-O. et VIALARD, A. «Discrete deformable boundaries for the segmentation of multidimensional images». Dans C. Arcelli, L. P. Cordella et G. Sanniti di Baja, éditeurs, *Proc. 4th Int. Workshop on Visual Form (IWVF4), Capri, Italy*, volume 2059 de *Lecture Notes in Computer Science*, pages 542–551. Springer-Verlag, Berlin. **2001**.
- LENOIR, A. *Des outils pour les surfaces discrètes : estimation d'invariants géométriques, préservation de la topologie, tracé de géodésiques et visualisation*. Thèse de doctorat, Université de Caen. **1999**.
- LENOIR, A., MALGOUYRES, R. et REVENU, M. «Fast Computation of the Normal Vector Field of the Surface of a 3D Discrete Object». *6th Discrete Geometry for Computer Imagery*, pages 101–112. **1996**.
- LINDBLAD, J. et NYSTRÖM, I. «Surface Area Estimation of Digitized 3D Objects Using Local Computations». Dans A. V. A. Braquelaire, J.-O. Lachaud, éditeur, *10th Discrete Geometry for Computer Imagery*, pages 267–278. Springer, LNCS, 2301, Bordeaux, France. **02**.
- LINDENBAUM, M. et BRUCKSTEIN, A. M. «On Recursive, $O(n)$ Partitioning of a Digitized Curve into Digital Straight Segments». *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(9):949–953. **1993**.
- LINDENBAUM, M. et KOPLOWITZ, J. «A New Parameterization of Digital Straight Lines». *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(8):847–852. **1991**.
- LOHOU, C. et BERTRAND, G. «A new 3D 6-Subiteration Thinning Algorithm based on P-Simple Points». Dans A. V. A. Braquelaire, J.-O. Lachaud, éditeur, *10th Discrete Geometry for Computer Imagery*, pages 102–113. Springer, LNCS, 2301, Bordeaux, France. **2002**.
- LORENSEN, W. E. et CLINE, H. E. «Marching Cubes: a High Resolution 3D Surface Construction Algorithm». Dans M. C. Stone, éditeur, *SIGGRAPH '87 Conference Proceedings (Anaheim, CA, July 27–31, 1987)*, pages 163–170. Computer Graphics, Volume 21, Number 4. **1987**.
- MALGOUYRES, R. «A definition of surfaces of \mathbb{Z}^3 : a new 3D discrete Jordan theorem». *Theoretical Computer Science*, 186(1–2):1–41. **1997**.
- MAURER JR., C., RAGHAVAN, V. et QI, R. «A Linear Time Algorithm for Computing the Euclidean Distance Transform in Arbitrary Dimensions». Dans *Information Processing in Medical Imaging*, pages 358–364. **2001**.
- MCILROY, M. D. «A Note on Discrete Representation of Lines». *AT&T Technical Journal*, 64(2):481–490. **1985**.

- MEGIDDO, N. «Linear-Time Algorithms for Linear Programming in \mathbb{R}^3 and Related Problems.» *SIAM Journal of Computing*, 12(4):759–776. **1983**.
- MEGIDDO, N. «Linear Programming in Linear Time When the Dimension Is Fixed.» *Journal of the ACM*, 31(1):114–127. **1984**.
- MEIJSTER, A., ROERDINK, J. B. T. M. et HESSELINK, W. H. «A general algorithm for computing distance transforms in linear time». Dans L. V. J. Goutsias et D. S. Bloomberg, éditeurs, *Mathematical Morphology and its Applications to Image and Signal Processing*, pages 331–340. Kluwer. **2000**.
- MESMOUDI, M. M. «A simplified recognition algorithm of digital planes pieces». Dans *10th International Conference, Discrete Geometry for Computer Imagery*, numéro 2301 dans LNCS, pages 404–416. Springer Verlag, Bordeaux, France. **2002**.
- MIGUET, S. et NICOD, J.-M. «Complexity analysis of a parallel implementation of the Maching-Cubes algorithm». *International Journal of Pattern Recognition and Artificial Intelligence*, 11(7):1141–1156. **1997**.
- MONGA, O., LENGAGNE, R. et DERICHE, R. *Crest lines in volume 3D medical images : a multi-scale approach*. Rapport technique, INRIA, Projet SYNTIM. **1994**.
- MONTANARI, U. «Continuous skeletons from digitized images». *Journal of the Association for Computing Machinery*, 16(4):534–549. **1969**.
- MOREAU, P. *Modélisation et génération de dégradés dans le plan discret*. Thèse de doctorat, Université Bordeaux I. **1995**.
- MORGENTHALER, D. G. et ROSENFELD, A. «Surfaces in Three-Dimensional Digital Images». *Information and Control*, 51(3):227–247. **1981**.
- MULLIKIN, J. C. «The Vector Distance Transform in Two and Three Dimensions». *Computer Vision, Graphics, and Image Processing. Graphical Models and Image Processing*, 54(6):526–535. **1992**.
- MULLIKIN, J. C. et VERBEEK, P. W. «Surface area estimation of digitized planes». *Bioimaging*, 1:6–16. **1993**.
- NAKAMURA, A. et AIZAWA, K. «Digital Circles». *Computer Vision, Graphics, and Image Processing*, 26(2):242–255. **1984**.
- O’ROURKE, J. «An on-line algorithm for straight lines between data ranges». *Communication of the ACM : Programming Techniques and Data Structures*, 24(9):574–578. **1981**.
- PAPIER, L. *Polyédrisation et visualisation d’objets discrets tridimensionnels*. Thèse de doctorat, Université Louis Pasteur, Strasbourg, France. **1999**.
- PAPIER, L. et FRANÇON, J. «Evaluation de la normale au bord d’un objet discret 3D». *Revue de CFAO et d’informatique graphique*, 13:205–226. **1998**.
- PERROTON, L. *Segmentation parallèle d’images volumiques*. Thèse de doctorat, Ecole Normale Supérieure de Lyon. **1994**.
- PIPER, J. et GRANUM, E. «Computing Distance Transformations in Convex and Non-Convex Domains». *Pattern Recognition*, 20:599–615. **1987**.

- PITTEWAY, M. L. V. «Integer circles, etc. - some further thoughts». *Computer Graphics and Image Processing*, 3:262–265. **1974**.
- PREPARATA, F. P. et SHAMOS, M. I. *Computational Geometry : An Introduction*. Springer-Verlag. **1985**.
- RAGNEMALM, I. *Contour processing distance transforms*, pages 204–211. World Scientific. **1990**.
- REMY, E. *Normes de chanfrein et axe médian dans le volume discret*. Thèse de doctorat, Université de la Méditerranée (Aix-Marseille 2). **2001**.
- REMY, E. et THIEL, E. «Optimizing 3D Chamfer Masks with Norm Constraints». Dans *Int. Workshop on Combinatorial Image Analysis*, pages 39–56. Caen. **2000**.
- REVEILLÈS, J.-P. *Géométrie discrète, calcul en nombres entiers et algorithmique*. Thèse de doctorat, Université Louis Pasteur, Strasbourg. **1991**.
- REVEILLÈS, J.-P. «The geometry of the intersection of voxel spaces». Dans S. Fourey, G. T. Herman et T. Y. Kong, éditeurs, *International Workshop on Combinatorial Image Analysis*, volume 46 de *Electronic Notes in Theoretical Computer Science*. Elsevier Science Publishers, Temple University, Philadelphia, U.S.A. **2001**.
- REVEILLÈS, J.-P. et YAACOUB, G. «A sublinear 3D convexhull algorithm for lattices». Dans *5th Discrete Geometry for Computer Imagery*, pages 219–230. **1995**.
- ROSENFELD, A. «Connectivity in Digital Pictures». *Journal of the ACM*, 17(1):146–160. **1970**.
- ROSENFELD, A. «Digital straight lines segments». *IEEE Transactions on Computers*, pages 1264–1369. **1974**.
- ROSENFELD, A. et KLETTE, R. «Digital Straightness». Dans S. Fourey, G. Herman et T. Kong, éditeurs, *International Workshop on Combinatorial Image Analysis*, volume 46 de *Electronic Notes in Theoretical Computer Science*. Elsevier Science Publishers, Temple University, Philadelphia, U.S.A. **2001**.
- ROSENFELD, A. et PFALTZ, J. L. «Sequential Operations in Digital Picture Processing». *Journal of the ACM*, 13(4):471–494. **1966**.
- ROSENFELD, A. et PFALTZ, J. L. «Distance functions on digital pictures». *Pattern Recognition*, 1:33–61. **1968**.
- SAITO, T. et TORIWAKI, J. I. «New algorithms for Euclidean distance transformations of an n -dimensional digitized picture with applications». *Pattern Recognition*, 27:1551–1565. **1994**.
- SALAM, I., NEHLIG, P. et ANDRÈS, E. «Discrete Ray-Casting». *Discrete Geometry for Computer Imagery*. **1999**.
- SANNITI DI BAJA, G. et SVENSSON, S. «Detecting centres of maximal discs». *9th Discrete Geometry for Computer Imagery*, pages 443–452. **2000**.
- SAUER, P. «On the recognition of digital circles in linear time». *Computational Geometry: Theory and Application 2*, pages 287–302. **1993**.

- SETHIAN, J. A. «A Fast Marching Level Set Method for Monotonically Advancing Fronts». *Proc. Nat. Acad. Sci.*, 93:1591–1595. **1996a**.
- SETHIAN, J. A. *Level Set Methods*. Cambridge University Press. **1996b**.
- SINTORN, I.-M. et BORGEFORS, G. «Weighted Distance Transforms for Images Using Elongated Voxel Grids». Dans *10th International Conference on Discrete Geometry for Computer Imagery*, numéro 2301 dans LNCS, pages 244–254. Springer. **2002**.
- SIVIGNON, I. *Reconnaissance de plans discrets et polyédration d'objets 3D*. Ecole Normale Supérieure de Lyon, Mémoire de DEA au Laboratoire des Images et des Signaux, Grenoble. **2001**.
- SIVIGNON, I. et COEURJOLLY, D. «From digital plane segmentation to polyhedral representation». Dans *Theoretical Foundations of Computer Vision, Geometry, Morphology, and Computational Imaging*. Springer-Verlag, LNCS. A paraître. **2002**.
- SLOBODA, F. et ZATKO, B. *On one-dimensional grid continua in \mathbb{R}^2* . Rapport technique, Institute of Control Theory and Robotics, Bratislava. **1996**.
- SLOBODA, F. et ZATKO, B. «On approximation of Jordan surfaces in 3D». Dans G. Bertrand, A. Imiya et R. Klette, éditeurs, *Digital and Image Geometry*, volume 2243 de LNCS, pages 365–388. Springer-Verlag. **2001**.
- SLOBODA, F., ZATKO, B. et FERIANC, P. *Advances in Digital and Computational Geometry*, chapitre On approximation of planar one-dimensional continua. Springer. **1998**.
- SOILLE, P. «Spatial distributions from contour lines: an efficient methodology based on distance transformations». *Journal of Visual Communication and Image Representation*, 2(2):138–150. **1991**.
- SOILLE, P. «Generalized geodesy via geodesic time». *Pattern Recognition Letters*, 15(12):1235–1240. **1994**.
- SOILLE, P. *Morphological Image Analysis*. Springer-Verlag, Berlin, Heidelberg, New York. **1999**.
- STOJMENOVIĆ, I. et TOSIĆ, R. «Digitization schemes and the recognition of digital straight lines, hyperplanes and flats in arbitrary dimensions». Dans *Vision Geometry, contemporary Mathematics Series*, volume 119, pages 197–212. American Mathematical Society, Providence, RI. **1991**.
- TEH, C.-H. et CHIN, R. «On the Detection of Dominant Points on Digital Curves». *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11:859–872. **1989**.
- TELLIER, P. et DEBLED-RENNESON, I. «3D Discrete Normal Vector». Dans *8th Discrete Geometry for Computer Imagery*, pages 447–458. Springer-Verlag, LNCS, 1568. **1999**.
- THIEL, E. *Les distances de chanfrein en analyse d'images : fondements et applications*. Thèse de doctorat, Université Joseph Fourier. **1994**.

- THIEL, E. *Géométrie des distances de chanfrein*. Habilitation à diriger les recherches, Université de la Méditerranée (Aix-Marseille 2). **2001**.
- THIRION, J. P. et GOURDON, A. *The 3D Marching Lines Algorithm and its Application to Crest Lines Extraction*. Rapport technique 1672, INRIA. **1992**.
- THOMASSEN, C. «On the Complexity of Finding a Minimum Cycle Cover of a Graph». *SIAM Journal on Computing*, 26(3):675–677. **1997**.
- THÜRMER, G. «Smoothing Normal Vectors on Discrete Surfaces while Preserving Slope Discontinuities». *Computer Graphics Forum*, 20(2):103–114. **2001**.
- THÜRMER, G. et WÜTHRICH, C. A. «Varying neighbourhood parameters for computation of normals on surfaces in discrete space». Dans *Proc. Computer Graphics International*, pages 616–625. IEEE Computer Society Press. **1998**.
- TOUGNE, L. *Cercles discrets sur automates cellulaires*. Thèse de doctorat, École Normale Supérieure de Lyon. **1997**.
- TROESCH, A. «Interpretation géométrique de l’algorithme d’Euclide et reconnaissance de segments». *Theoretical Computer Science*, 115(2):291–319. **1993**.
- UDUPA, J. K. «Multidimensional Digital Boundaries». *Computer Vision and Image Processing: Graphical models and image processing*, 56:311–323. **1994**.
- UDUPA, J. K., SRIHARI, S. N. et HERMAN, G. T. «Boundary detection in multi-dimensions». *IEEE Pattern Analysis and Machine Intelligence*, 4(1):41–50. **1982**.
- VAN DER WAERDEN, B. L. *Geometry and Algebra in Ancient Civilizations*. Springer. **1983**.
- VEELAERT, P. «On the Flatness of digital hyperplanes». *Journal of Mathematical Imaging and Vision*, 3:205–221. **1993**.
- VEELAERT, P. «Digital planarity of rectangular surface segments». *IEEE Pattern Analysis and Machine Intelligence*, 16(6):647–652. **1994**.
- VERWER, B. J. H. «Local distances for distance transformations in two and three dimensions». *Pattern Recognition Letters*, 12:671–682. **1991**.
- VERWER, B. J. H., VERBEEK, P. W. et DEKKER, S. T. «An Efficient Uniform Cost Algorithm Applied to Distance Transforms». *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(4):425–429. **1989**.
- VIALARD, A. *Chemin euclidiens : un modèle de représentation des contours discrets*. Thèse de doctorat, Université de Bordeaux 1. **1996a**.
- VIALARD, A. «Geometrical Parameters Extraction from Discrete Paths». Dans *6th Discrete Geometry for Computer Imagery*, pages 24–35. Springer-Verlag, LNCS, 1176. **1996b**.
- VIDAL, B. *Vers un lancer de rayon discret*. Thèse de doctorat, Thèse de Doctorat, Lille. **1992**.
- VITTONI, J. *Caractérisation et reconnaissance de droites et de plans en géométrie discrète*. Thèse de doctorat, Université Joseph Fourier-Grenoble 1. **1999**.

- VITTONÉ, J. et CHASSÉRY, J.-M. «Coexistence of tricubes in digital naive plane». Dans *7th International Workshop in Discrete Geometry for Computer Imagery*, pages 99–110. Springer-Verlag, LNCS, 1347. **1997**.
- VITTONÉ, J. et CHASSÉRY, J.-M. «Recognition of Digital Naive Planes and Polyhedrization». Dans *Discrete Geometry for Computer Imagery*, numéro 1953 dans LNCS, pages 296–307. Springer. **2000**.
- VOSS, K. *Discrete Images, Objects and Functions in \mathbb{Z}^n* . Numéro 11 dans Algorithms and Combinatorics. Springer-Verlag. **1993**.
- VOSSEPOEL, A. M. et SMEULDERS, A. W. M. «Vector Code Probability and Metrication Error in the Representation of Straight Lines of Finite Length». *Computer Graphics and Image Processing*, 20(4):347–364. **1982**.
- WEST, D. B. *Introduction to Graph Theory*. Prentice Hall, seconde édition. **2001**.
- WORRING, M. et SMEULDERS, A. W. M. «Digital Curvature Estimation». *Computer Vision, Graphics, and Image Processing. Image Understanding*, 58(3):366–382. **1993**.
- WORRING, M. et SMEULDERS, A. W. M. «Digitized circular arcs: characterization and parameter estimation». *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(6):587–598. **1995**.
- WU, L. D. «On the chain code of a line». *IEEE Trans. Pattern Analysis and Machine Intelligence*, 4:347–353. **1982**.
- YAGEL, R., COHEN, D. et KAUFMAN, A. «Normal estimation in 3d discrete space». *The Visual Computer*, 8:278–291. **1992**.
- YAMADA, H. «Complete Euclidean Distance Transformation by Parallel Operation». Dans *Seventh International Conference on Pattern Recognition*, pages 69–71. IEEE Publications. **1984**.
- YU, D. et KLETTE, R. «An approximative calculation of relative convex hulls for surface area estimation of 3D digital objects». Dans *International Conference on Pattern Recognition*. IEEE Computer Science, Quebec. **2002**.